ACKSYS

MI-ETH 2/4/8 PORTS

USER MANUAL
Version 1.3

ACKSYS COMMUNICATIONS & SYSTEMS
3-5 rue du Stade
BP 4580
F 78302 POISSY CEDEX
FRANCE
Tel : +33 (0)1 39 11 62 81
Fax : +33 (0)1 39 11 29 30
www.acksys.fr

I. INTRODUCTION 6

I.1 – REMOTE ACCESS SERVER AND SERIAL PORTS SERVER'S SERIE FAMILY 6
I.2 – FURNISHED FEATURES 6
I.3 - SYMBOLICS 6

II. MI-ETH OVERVIEW 8

II.1 - MI-ETH 2 PORTS 8
II.2 - MI-ETH 4 & 8 PORTS 8

III.INSTALLATION 10

III.1 – INDICATING LEDS OF MI-ETH 2 10
III.2 – INDICATING LEDS OF MI-ETH 4 & 8 PORTS 11
III.3 - PORTS AND INTERFACES OF MI-ETH 2 PORTS 11
III.4 - PORTS AND INTERFACES OF MI-ETH 4 & 8 PORTS 11
III.5 - CAUTIONS 12
III.5.1 – Cleaning MI-ETH 12
III.5.2 - Note 12
III.5.2.1 - About MI-ETH 2 ports 12
III.5.2.2 – General note 12

IV. CONFIGURATION 13

IV.1 - INTRODUCTION 13
IV.2 – ENTERING IP ADRESS 13
IV.2.1 - Entering IP adress from Advanced Administrator 13
IV.2.2 - Entering IP adress from a terminal 14
IV.3 - HTTP CONFIGURATION (NETSCAPE, INTERNET EXPLORER) 14
IV.4 – NETWORK INTERFACE 16
IV.4.1 - Identification 16
IV.4.2 – Rout checking with PING command 17
IV.4.4 – Routing table 17
IV.5 – ASYNCHRONOUS INTERFACES 19
IV.5.1 – Software settings 19
IV.5.2 - Asynchronous port mode 19
IV.5.3 - Type of control 20
IV.5.4 – Flow Control 20
IV.5.5 – Automatic disconnection of a Modem 21
IV.5.6 - Association of a modem form to a port 21
IV.5.7 – AutoUser mode 21
IV.6 – SAVING AND RESTORING CONFIGURATION 22
IV.6.1 – Saving in FLASH memory 22
IV.6.2 – Saving a configuration copy 22
IV.7 – OPENING OF A USER ACCOUNT 22
IV.7.1 – Name and password 23
IV.7.2 - Comments 23
IV.7.3 - In and Out Communication 23
IV.7.4 - Callback 23
IV.7.5 - Audit 25
IV.7.6 - Optional remote IP Address 25
IV.7.7 - Telnet Address 25

IV.8 – MANAGEMENT OF A MODEM CONNECTION 26
IV.9 - AUDIT 27
IV.9.1 - Audit on a remote machine *27*
IV.9.2 - Audit on your console *28*
IV.9.3 – Audit in a file *28*
IV.9.4 – List of created audits *28*
IV.9.5 - Delete audit *28*
IV.9.6 – Start / stop audit *29*
V.TERMINAL MODE (TELNET) 30
V.1 - INTRODUCTION 30
V.2 – TELNET 30
V.3 – MULTI-SESSIONS 31
V.4 – AUTOMATIC TELNET 32
V.5 – USER CONNECTION DIAGRAM 32
VI. RAWTTY AND REMOTECOM 33
VI.1 – INTRODUCTION 33
VI.2 – REMOTECOM UNDER WINDOWS 95/98 & NT 33
VI.2.1 – Setting up the MI-ETH *33*
VI.2.2 - Installation of RemoteCOM driver *33*
VI.3 - RAWTTY 37
VI.3.1 – Server configuration *37*
VI.3.2 – Setting up MI-ETH *37*
VI.3.3 - rawtty, rawclose *37*
VII.MUX MODE 39
VII.1 - INTRODUCTION 39
VII.2 – TCP MUX 39
VII.3 – UDP MUX 39
VII.4 - EXAMPLE N°1: REMOTE MANAGEMENT OF A PLC BY A TECHNICIAN 39
VII.4.1 - Description *39*
VII.4.2 - Configuration of client's side MI-ETH *40*
VII.4.3 - Configuration of server's side MI-ETH *41*
VII.5 - EXAMPLE N°2: REMOTE MANAGEMENT OF A PLC BY MORE THAN ONE TECHNICIAN (SOLUTION 1) 41
VII.5.1 - Description *41*
VII.5.2- Solution *42*
VII.5.3 - Configuration of client's side MI-ETH *42*
VII.5.4 - Configuration of server's side MI-ETH *43*
VII.6 - EXAMPLE N°3: REMOTE MANAGEMENT OF A PLC BY MORE THAN ONE TECHNICIAN (SOLUTION 2) 44
VII.6.1 - Description *44*
VII.6.2 – Solution *44*
VII.6.3 - Configuration of client's side MI-ETH *45*
VII.6.4 - Configuration of server's side MI-ETH *46*
VII.7 - EXAMPLE N°4: REMOTE MANAGEMENT OF A PLC BY MORE THAN ONE TECHNICIAN (SOLUTION 3) 46
VII.7.1 – Description *46*
VII.7.2 – Solution *47*
VII.7.3 - Configuration of client's side MI-ETH *48*

VII.7.4 - Configuration of server's side MI-ETH *49*
VIII. PRINTER MODE (LPD) **50**
IX. RTELNET MODE **51**
IX.1 - DEFINITION 51
IX.2 - CONFIGURATION 51
IX.3 - RTELNET POOL 52
X. ASYNCHRONE PPP MODE **53**
X.1 - INTRODUCTION 53
X.2 - PAP AND CHAP AUTHENTICATION 53
X.3 - PPP CONNECTION SETUP 53
X.4 - DIAL-IN CONNECTION 53
X.5 - - PPP FORM 54
X.5.1 – Creation, cancelling and displaying of a form *54*
X.5.2 - Comments *54*
X.5.3 - Security *54*
X.5.4 - Authname and password *55*
X.5.5 - Local IP *55*
X.5.6 - Remote IP *55*
X.5.7 - Proxy ARP *55*
X.5.8 - Mask *56*
X.5.9 - Routing *56*
X.5.10 - Asyncmap *56*
X.5.11 - MRU *56*
X.5.12 - MTU *57*
X.6 – PPP CONNECTION ESTABLISHMENT 57
X.6.1 - Dial-IN user connection phase *57*
X.6.2 – Authentication phase *57*
X.6.3 – IP adress negotiation phase *58*
X.7 - COMPRESSION 59
X.8 - PROXY-ARP 59
X.9 - EXAMPLE N°1 : REMOTE ACCESS SERVER 60
X.9.1 - Configuration *60*
X.9.2 – Configuration of Windows 98/95 Dial-Up PPP Client. *62*
X.9.3 – Resources sharing on WINDOWS 95/98/NT *62*
X.10 – ACCESS SERVER FUNCTION WITHOUT PROXY-ARP 63
XI.ADVANCED FONCTIONS **64**
XI.1 – FIREWALL 64
XI.1.1 - Definition *64*
XI.1.2 - Chains *64*
XI.1.2.1 – Examples 65
XI.1.3 - Rules *66*
XI.1.3.1 – Bidirectionnal option 67
XI.1.3.2 - Type Of Service (TOS) 67
XI.1.4 – Examples *67*
XI.1.4.1 - Forward Firewall 68
XI.1.4.2 - Input Firewall 69
XI.2 – RADIUS 70
XI.2.1 - Introduction *70*

XI.2.2 - Attributs Radius supportés par le MI-ETH *71*
XI.2.3 – Radius requests *72*
XI.2.4 - Configuration of thr Radius client on MI-ETH *73*
XI.3 - DHCP 73
XI.3.1 - Definition *73*
XI.3.2 - DHCP client *74*
XI.3.3 – DHCP relay *75*
XII.TECHNICAL SPECIFICATIONS 75
XIII.CABLING 79
XIII.1 - SUBD9M CONNECTOR ON MIETH 2 PORTS 79
XIII.2 - RJ45 CONNECTOR ON MIETH 4 & 8 PORTS 79
XIII.3 - MI-ETH (DTE) TO DTE 80
XIII.4 - MI-ETH (DTE) TO ASYNCHRONOUS TERMINAL (DTE) 80
XIII.5 - MI-ETH (DTE) TO DCE EQUIPMENT - RS232C (MODEM) 81
XIII.6 - MI-ETH (DTE) TO DCE EQUIPMENT - RS422A 82
XIII.7 - MI-ETH (DTE) TO DCE EQUIPMENT - RS485 83

I. Introduction

I.1 – Remote access server and serial ports server's serie family

MI-ETH ensures two main functionalities, the functionalities of terminal server and of serial port server.

A terminal server enables you to connect passive terminals on a TCP/IP network. Thanks to MI-ETH's integrated Telnet client, every terminal will be able to open up to three sessions on the server.
Thanks to MI-ETH's remote access server function, you can perform a dial-in PPP connection on each MI-ETH's port. It makes possible for a PPP compatible portable (using a modem) to connect to the company's network.

A serial ports server provides real COM ports on Ethernet networks. You can use modems, barcodes readers or any serial device on the Ethernet network.

Thanks to the Mux mode, you can also establish a communication between two serial devices through an asynchronous link. You can, for example, use a terminal on your MI-ETH to manage a PLC located on a remote MI-ETH.

Remote access server's family is composed by the following products :

- MI-ETH 2 provides 2 asynchronous ports (RS232C) on SUBD9 and one Ethernet port on RJ45. Power supply is about 207 to 235 VAC.
  It integrates an electric circuit referred ACS 31 or ACS 32.
- MI-ETH 4 & 8 provides 4 or 8 asynchronous ports (RS232C or RS422/485A) on RJ45 and one Ethernet port on RJ45. Power supply is about
  84 to 264 VAC. It integrates an electric circuit referred ACS 320.

I.2 – Furnished features

You will find the following features with any MI-ETH :
- CD-ROM contains RemoteCOM drivers, Rawtty drivers and a few usefull softs.
Please read the readme.html file for more information.
- A power supply cable for 4 or 8 ports version only.

I.3 - Symbolics

In all pictures of this documentation, we will use the following symbols :

	WAN Link		Printer
	Ethernet Network		Modem
	Lan link		Aquisition de données
	Work		Portable computer
	Serial port server function / Raw Mode		Gateway
	Terminal server function		Remote access router function
	Mux Mode function		Server

II. MI-ETH overview

II.1 - MI-ETH 2 ports

MI-ETH 2 provides 2 asynchronous ports on which you can connect passive terminals.
Thanks to MI-ETH's integrated Telnet client, every terminal will be able to open up to three sessions on the server.
Thanks to MI-ETH's remote access server function, you can perform a dial-in PPP connection on each MI-ETH's port. It makes possible for a PPP compatible portable (using a modem) to connect to the company's network.

Thanks to the "raw" function of MI-ETH, you can connect printers or any other serial equipment (bar-code reader, data acquisition system, a.s.o...).
Just run on your Unix server the "rawtty" application to redirect the in and out data flow of the specified communication port ('/dev/ttyx') to one of MI-ETH port.
RemoteCOM is provides the same function for Windows 95/98/NT systems.

Thanks to the Mux mode, you can also establish a communication between to serial equipments through an asynchronous link. You can, for example, use a terminal on your MI-ETH to manage a PLC located on a remote MI-ETH.

II.2 - MI-ETH 4 & 8 ports

MI-ETH 4& 8 supports the same functions as MI-ETH 2 but provides 4 or 8 asynchronous ports.
With MI-ETH 4 & 8, you can use RemoteCOM function on each port.

Figure 1 - Using of MI-ETH as a terminal server.

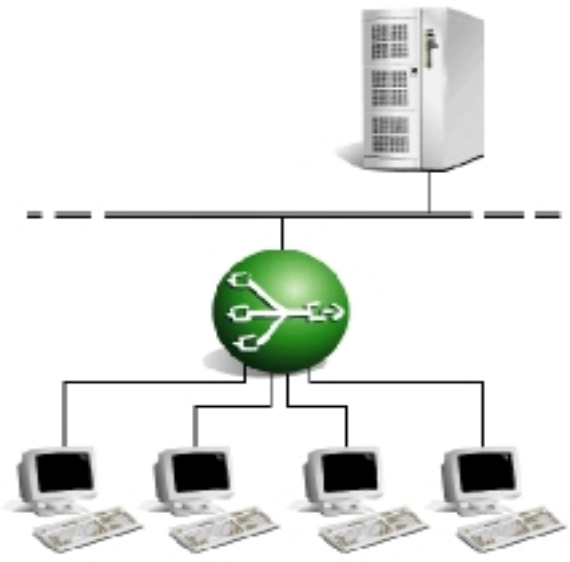


Figure 2 - Using of MI-ETH as remote access server

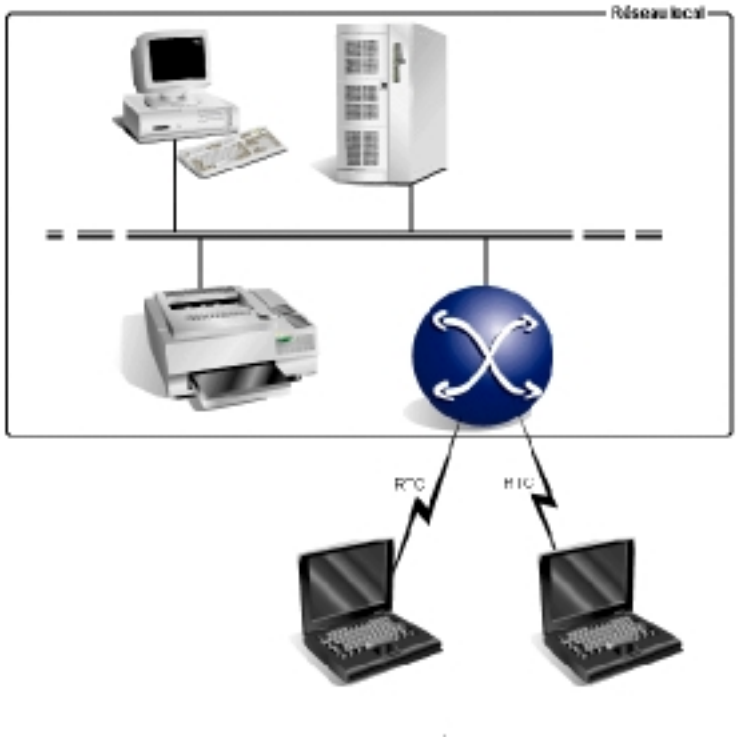
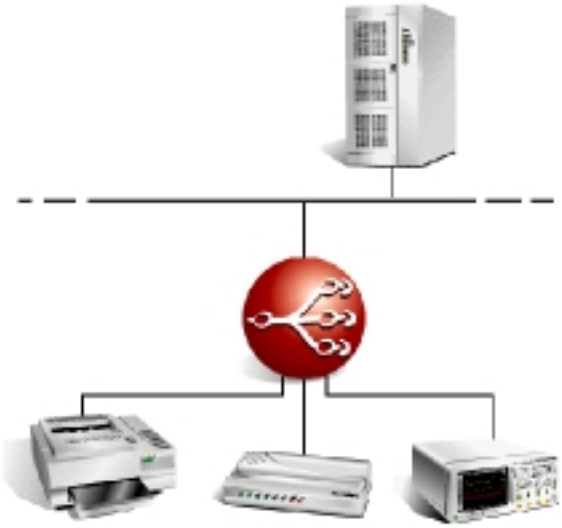


Figure 3 - Using of MI-ETH to connect serial equipments.



III.Installation

This chapter details the first steps to install MI-ETH 2 ports, and 4 & 8 ports

III.1 – Indicating LEDS of MI-ETH 2

Designation	Color	State	Meaning
Setup	yellow	On	Setup mode activated
		Off	Normal mode
		Blinking	Locate function activated
Link (eth)	yellow	On	No presence of the network
		Off	Presence of the network
Tx (eth)	green		Data transmission To the network
Rx (eth)	green		Data reception from the network
Tx (S1)	yellow		Data transmission To port S1
Rx (S1)	yellow		Data reception from port S1
Tx (S2]	yellow		Data transmission To port S2
Rx (S2)	yellow		Data reception from port S2
Power	red	On	MI_ETH is on
		Off	MI_ETH is off

III.2 – Indicating LEDS of MI-ETH 4 & 8 ports

Designation	Color	State	Meaning
Setup	red	On	Setup mode activated
		Off	Normal mode
		Blinking	Locate function activated
LAN Link	red	On	No presence of the network
		Off	Presence of the network
LAN Tx	green		Data transmission from the network
LAN Rx	green		Data reception from the network
Power	green	On	MI_ETH is on
		Off	MI_ETH is off

III.3 - Ports and interfaces of MI-ETH 2 ports

The rear face of MI-ETH 2 displays the followings elements :

- 1 power supply connector.
- 1 Ethernet port 10baseT
- 2 asynchronous ports RS232C/V24 on SUBD9 noted ’’Serial 1” and “serial 2”.

-. The port n°1 is the console port in setup mode

The front face of MI-ETH 2 displays the following elements :
- Indicating LEDS.

III.4 - Ports and interfaces of MI-ETH 4 & 8 ports

The rear face of MI-ETH 4 & 8 displays the followings elements :
- 1 power supply connector.
- 1 Ethernet port 10baseT.
- 4 or 8 asynchronous ports (RS232C/V24 or RS422/485A) on RJ45

The port n°1 is the console port in setup mode

The front face of MI-ETH 4 & 8 displays the following elements :
- Indicating LEDS.

III.5 - Cautions

The opening of the box is strictly reserved to persons authorized by Acksys.
Manipulation of the open box is forbidden.

III.5.1 – Cleaning MI-ETH

- Remove the power supply cable from the rear panel
- Please use watertown.

Caution : do not use chemical products which can damage the box paint (Alcohol, trichlo,..)

- Please be sure the box is completely dry before powering on.

III.5.2 - Note

III.5.2.1 - About MI-ETH 2 ports

Security of this equipment is guaranteed only if you use the furnished power supply or an equivalent model.

Warning

This is a class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

III.5.2.2 – General note

The hardware and software described in this document may change without warning.
The information in this document may change without warning. ACKSYS reserves itself the right to revise this publication without having to provide notification for such revisions.
Aslong as reasonable precautions have been taken, ACKSYS assumes no responsibility for errors that may appear in this document.
No part of this publication may be copied or reproduced in any form or by means without prior written consent of ACKSYS.
Windows and Microsoft Windows are trademarks of Microsoft Corporation. Ethernet is a trademark of the Xerox corporation. Netscape is a trademark of Netscape Communications Corporation.
All other brand and product names and trademarks mentioned herein are trademarks of their respective owners.

IV. Configuration

IV.1 - Introduction

The three ways to configure your MI-ETH are :
- With an asynchronous terminal (or a terminal emulator) connected to one of its serial ports.
- From another network host using Telnet protocol.
- From a web browser (Netscape, Internet Explorer,...).

Before using Telnet or HTTP protocols, you just need to fix IP address. This can be made with Administrator or with a terminal.

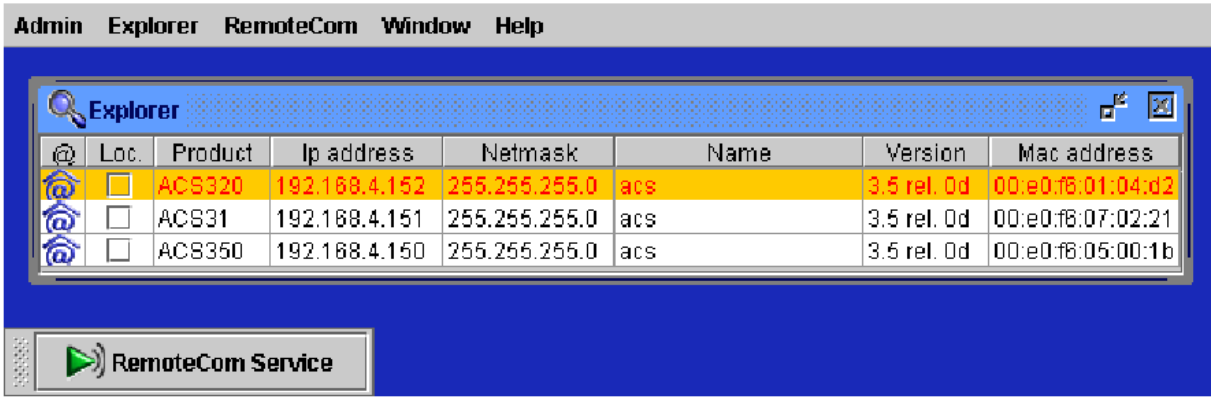
Before reaching the mode of configuration, a " login " and a "password " are required, it is about " ***root*** " and " ***root*** ".

IV.2 – Entering IP adress

IV.2.1 - Entering IP adress from Advanced Administrator

Advanced administrator enables you to locate and setup all MI-ETHs of your LAN.
To search for all MI-ETHs on your LAN, click on "Adminstration" then on "Explorer". A window like this one must appears on your screen :

(Warning: MIETH are recognized under the name of ACS 32 or 31 for the version 2 ports, and ACS320 for version 4 and 8 ports).



To change MI-ETH's IP address, click on "IP address" cell and type new IP adress.

IV.2.2 - Entering IP adress from a terminal

Use a terminal (9600 bauds, 8 bits, no parity) connected on MI-ETH's Console port (Port 1).
The Setup mode can be selected after MI-ETH startup. When the Setup LED blinks, press three times on '@' :
The terminal screen should display the following message :

```
ROOT>>
```

Now, you have access to a classic line shell (with a vt100, vt220, ansi terminal). Data input errors may be corrected by using 'left arrow ', 'right arrow' and 'backspace' keys. The 'up arrow' and 'down arrow' keys re-edit the last command.
You can now enter IP address and the network where your MI-ETH is connected, with the following commands :

IFCONFIG IP <interface> <IP>_address>
ROUTE ADD NET <interface> <destination> <gateway> <netmask>

Example :

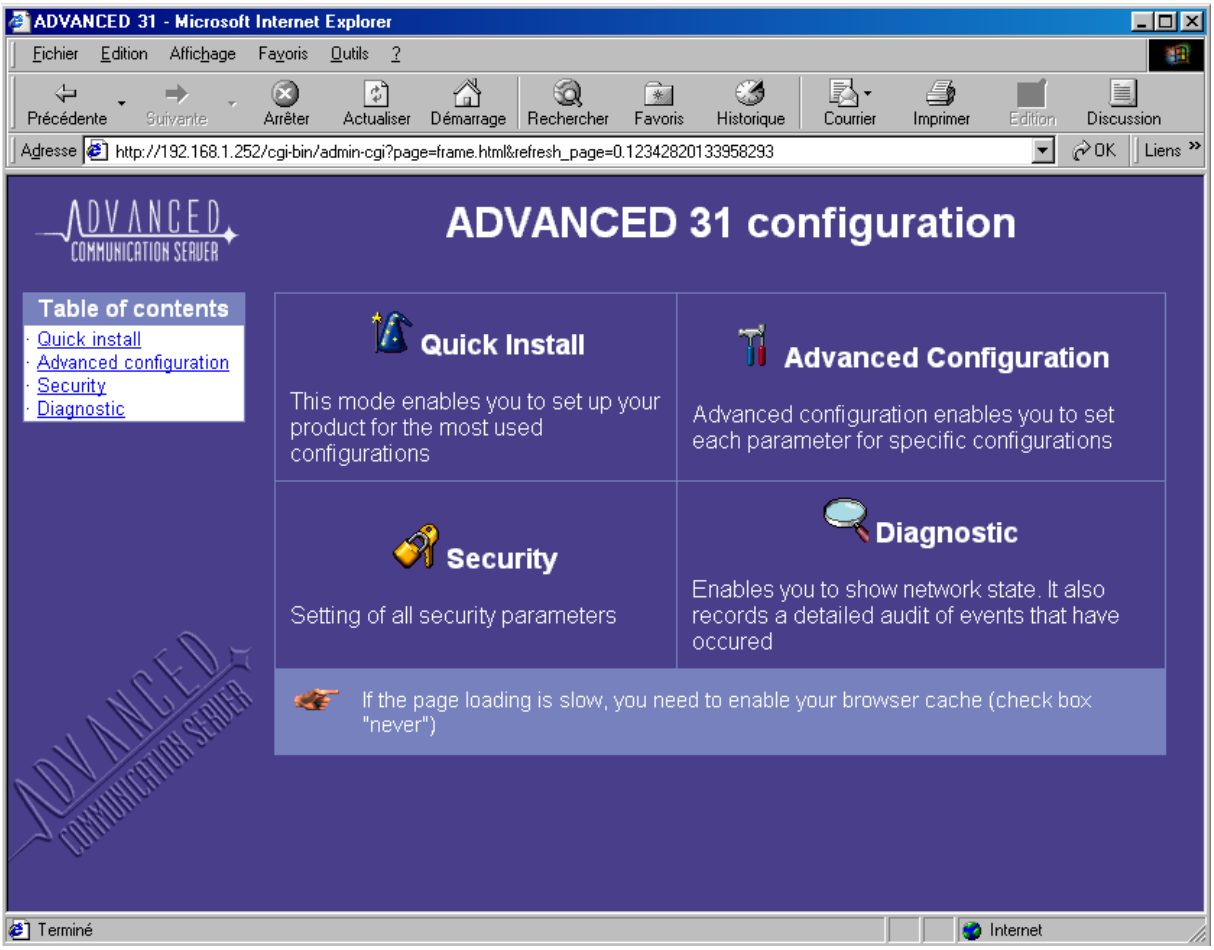
```
ROOT>> ifconfig ip eth0 192.168.1.11
Ok!
ROOT>> route add net eth0 192.168.1.0 0.0.0.0 255.255.255.0
Ok!
ROOT>> ifconfig ip eth1 192.168.2.22
Ok!
ROOT>> route add net eth1 192.168.2.0 0.0.0.0 255.255.255.0
Ok!
```

To save your configuration parameters, it is necessary to write it in permanent memory (FLASH-EPROM). Enter the **save** command to do achieve this :

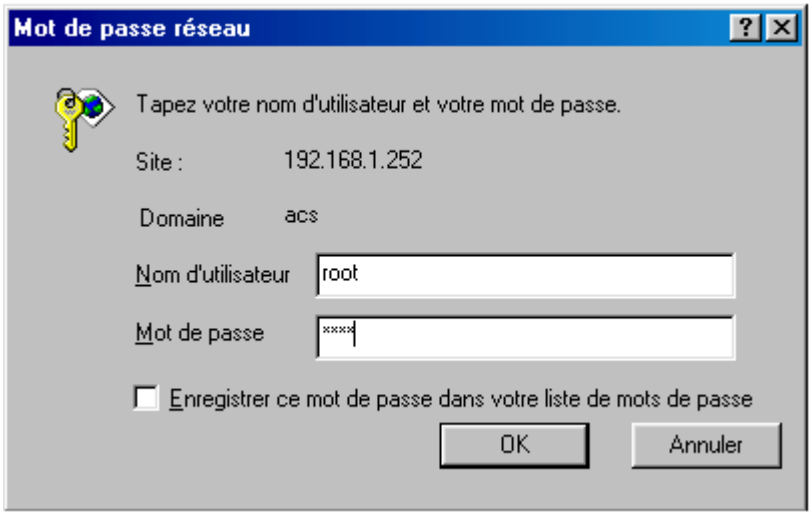
```
ROOT>> save
Status: Done.
ROOT>>
```

IV.3 - HTTP Configuration (Netscape, Internet Explorer)

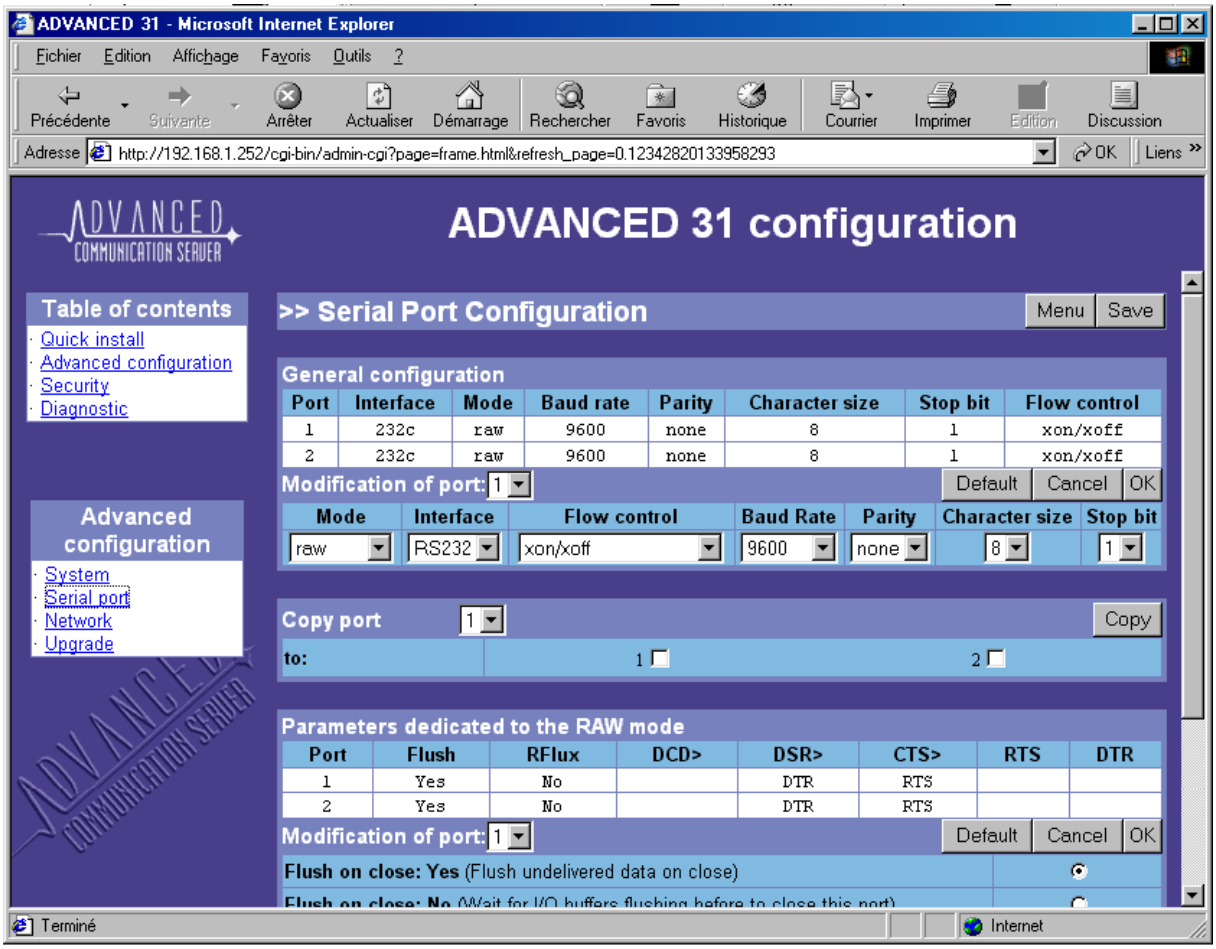
MI-ETH can be configured in text mode (Console, Telnet) or in graphic mode (HTTP).
The following pages of this manual excusively deal with a text mode configuration.
However, all text mode commands have their equivalents in graphic mode.
To set up parameters in graphic mode, you just need a web browser (Netscape, Internet Explorer), and to enter MI-ETH's IP address in URL field :


ADVANCED 31 - Microsoft Internet Explorer
Fichier Edition Affichage Favoris Outils ?
Précédente Suivante Arrêter Actualiser Démarrage Rechercher Favoris Historique Courrier Imprimer Edition Discussion
Adresse http://192.168.1.252/cgi-bin/admin-cgi?page=frame.html&refresh_page=0.12342820133958293
OK Liens
ADVANCED COMMUNICATION SERVER
ADVANCED 31 configuration
Table of contents
Quick install
Advanced configuration
Security
Diagnostic
Quick Install
This mode enables you to set up your product for the most used configurations
Advanced Configuration
Advanced configuration enables you to set each parameter for specific configurations
Security
Setting of all security parameters
Diagnostic
Enables you to show network state. It also records a detailed audit of events that have occured
If the page loading is slow, you need to enable your browser cache (check box "never")
Terminé
Internet

Click on GO, and open a session under the name root. Default password is root.


Mot de passe réseau
Tapez votre nom d'utilisateur et votre mot de passe.
Site : 192.168.1.252
Domaine acs
Nom d'utilisateur root
Mot de passe ****
Enregistrer ce mot de passe dans votre liste de mots de passe
OK
Annuler

Now you can configure your MI-ETH.


ADVANCED 31 - Microsoft Internet Explorer
Fichier Edition Affichage Favoris Outils ?
Précédente Suivante Arrêter Actualiser Démarrage Rechercher Favoris Historique Courrier Imprimer Edition Discussion
Adresse http://192.168.1.252/cgi-bin/admin-cgi?page=frame.html&refresh_page=0.12342820133958293
OK Liens
ADVANCED COMMUNICATION SERVER
ADVANCED 31 configuration
Table of contents
Quick install
Advanced configuration
Security
Diagnostic
Advanced configuration
System
Serial port
Network
Upgrade
>> Serial Port Configuration
Menu Save
General configuration
Port Interface Mode Baud rate Parity Character size Stop bit Flow control
1 232c raw 9600 none 8 1 xon/xoff
2 232c raw 9600 none 8 1 xon/xoff
Modification of port: 1
Default Cancel OK
Mode Interface Flow control Baud Rate Parity Character size Stop bit
raw RS232 xon/xoff 9600 none 8 1
Copy port 1
Copy
to: 1 2
Parameters dedicated to the RAW mode
Port Flush RFlux DCD> DSR> CTS> RTS DTR
1 Yes No DTR RTS
2 Yes No DTR RTS
Modification of port: 1
Default Cancel OK
Flush on close: Yes (Flush undelivered data on close)
Terminé
Internet

IV.4 – Network interface

IV.4.1 - Identification

To correctly initialize the network connection, use the **ifconfig** command to enter the following parameters :

- ***IP address***. The location of your MI-ETH on the network. This address consists in four numbers separated by dots (.), valid values for each of the four numbers are whole numbers between 1 and 254, for example *"192.168.2.*3".
- ***Network mask*** (netmask). It is a number similar to IP address, which determines what address is specified by the network and what place on this network is specified by your MI-ETH. For an IP address *"192.168.2.*3", if you enter *255.255.255.*0" as a mask value, this means that your network address is *"192.168.2.*0".
- ***The maximal transfer unit system*** (MTU). On an Ethernet system the MTU value is commonly established at 1500.
- ***Broadcast address***. By default, this address is the one of the system with 255 in place of the machine address. For a network address *"192.168.2.*0", broadcast address will be *"192.168.2.255*".

By default, network mask is configured at *"255.255.255.*0" (class C) and the MTU at 1500 (Ethernet). Only in some particular cases, you have to change these values.

IV.4.2 – Rout checking with PING command

In order to check if MI-ETH is can be reached inside the network, you can run the **ping** command :

```
ROOT>> ping 192.168.1.20                                adresse IP d'une machine du
réseau
PING 192.168.1.20 (192.168.1.20): 56 data bytes
64 bytes from 192.168.1.20: icmp_seq=0 ttl=255 time=0.6 ms
64 bytes from 192.168.1.20: icmp_seq=1 ttl=255 time=38.2 ms
64 bytes from 192.168.1.20: icmp_seq=2 ttl=255 time=2.7 ms
64 bytes from 192.168.1.20: icmp_seq=3 ttl=255 time=2.7 ms
64 bytes from 192.168.1.20: icmp_seq=4 ttl=255 time=2.8 ms

--- 192.168.1.20 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.6/9.4/38.2 ms

ROOT>>
```

IV.4.4 – Routing table

Before sending an IP frame, MI-ETH consults its routing table in order to find an entry indicating the route to use.
In the following example, terminals connect to server thanks to MI-ETH.
Note that the server is to be found on the sub-network *192.168.1.0* at the address *192.168.1.2* and that MI-ETH belongs to the sub-network *192.168.2.0* at the address *192.168.2.*2. A bridge helps to the interconnection of the two sub-networks :

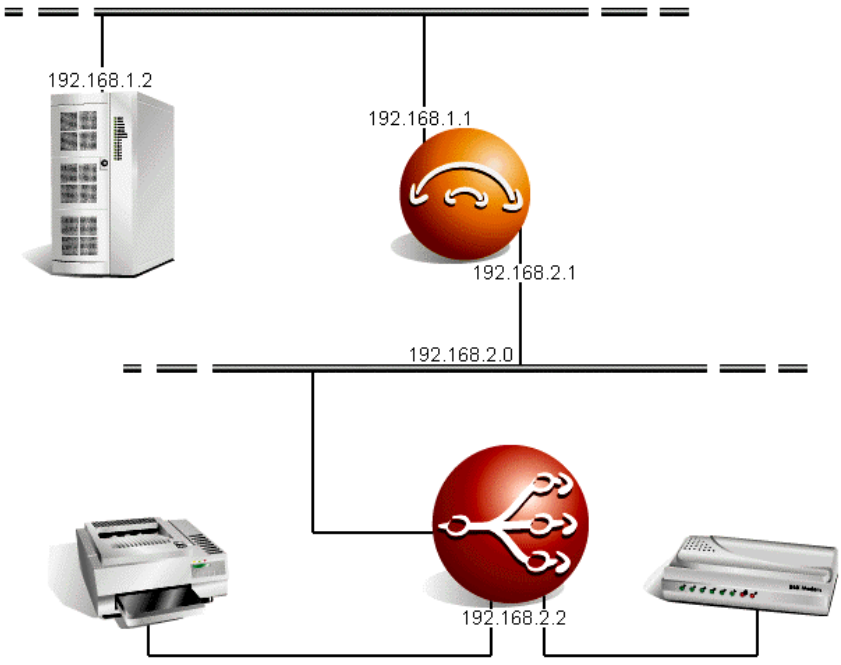


Figure 5 - Network example to make a routing table

To send IP frames to server, you must necessarily inform MI-ETH that these frames run through the *192.168.2.1* router.
There are three manners for this :
The first consists in informing MI-ETH that all IP frames intended for the *192.168.1.2* server have to be sent to the *192.168.2.1* router :

```
ROOT>> route add host eth0 192.168.1.2 192.168.2.1 255.255.255.255
Ok!
ROOT>> route static
Kernel routing table
Destination     Gateway       Genmask           Type     Iface
192.168.2.0     0.0.0.0       255.255.255.0      net     eth0
192.168.1.2     192.168.2.1   255.255.255.255    host    eth0
```

The second consists in informing MI-ETH that all IP frames intended for the *192.168.1.0* network have to be transmitted to the *192.168.2.1* router :

```
ROOT>> route add net eth0 192.168.1.0 192.168.2.1 255.255.255.0
Ok!
ROOT>> route static
Kernel routing table
Destination     Gateway       Genmask         Type     Iface
192.168.2.0     0.0.0.0       255.255.255.0    net     eth0
192.168.1.0     192.168.2.1   255.255.255.0    net     eth0
```

The third, (and most often used manner) consists in informing MI-ETH that, if the routing table does not contain any entry corresponding to the IP frame to send, then this one will be sent (by default) to the *192.168.2.1* router :

```
ROOT>> route add net eth0 0.0.0.0 192.168.2.1 0.0.0.0
Ok!
ROOT>> route static
Kernel routing table
Destination     Gateway       Genmask         Type     Iface
192.168.2.0     0.0.0.0       255.255.255.0    net     eth0
0.0.0.0         192.168.2.1   0.0.0.0          net     eth0
```

IV.5 – Asynchronous interfaces

IV.5.1 – Software settings

Configuration of a serial port can be done with **serial** command :
You must enter individually each parameter :

```
ROOT>> serial speed 9600 4
ROOT>> serial csize 8 4
ROOT>> serial parity none 4
ROOT>> serial stopb 1 4
```

Port 4 can then be setup to 9600 bauds, without parity control, with a 8 bits data width and one stop bit. To set the port speed of ports 3, 6 and 7 to 19200 bps, type the following command :

```
ROOT>> serial speed 19200 3 6 7
```

To set all ports to 38400 bps, enter the following command :

```
ROOT>> serial speed 38400 all
```

To set all ports but ports 3 and 5 to 4800 bps, enter :

```
ROOT>> serial speed 4800 all 3 5
```

You can see, at every time, all current ports parameters with the following command :

```
ROOT>> serial show all
```

IV.5.2 - Asynchronous port mode

Each asynchronous port can work in one of the following modes (regardless of the other ports configurations) :

-Terminal mode : Connection of an asynchronous terminal (Telnet).
-Printer mode : Connection of a printer (LPD).
-Raw mode : Connection of a passive equipment (RemoteCOM or rawTTY).
-PPP mode :
Connection of a remote host to a network or remote networks interconnection. (asynchronous PPP).
-Rtelnet mode :
Access through a TCP/IP network to an equipment or to an operating system not equiped with the TCP/IP protocol.

For example, to program the first five MI-ETH ports in each of these modes, commands would respectively be :

```
ROOT >> serial mode term 1
Ok!
ROOT >> serial mode printer 2
Ok!
ROOT >> serial mode raw 3
Ok!
ROOT >> serial mode ppp 4
Ok!
ROOT >> serial mode rtelnet 5
Ok!
```

IV.5.3 - Type of control

A port can be configured in *local control* or in *modem contro*l. In *modem contro*l, MI-ETH will ignore state of the DCD signal for this port. On the other hand, in *modem contro*l, MI-ETH will
propose to open a session only if the DCD is active. Moreover, if the DCD signal becomes inactive, MI-ETH will close all the open sessions on this port.
To configurate port 3 in *local contro*l, enter:

```
ROOT>> serial linectrl local 3
```

To use a modem on port 2, you have to setup the port with this command :

```
ROOT>> serial linectrl modem 2
```

IV.5.4 – Flow Control

Flow control can have four different values :

Ssoft control	XON/XOFF. Sending of XON(11H) and XOFF(13H) characters.
Hard control	RTS/CTS. Signal shift.
Soft and hard control	XON/XOFF and RTS/CTS. This kind of flow control is used when one want to connect a serial printer on a port.
No control	No flow control will be performed.

For example, to set up the first four ports of MI-ETH in each of these flow control modes, you have to enter the following commands :

```
ROOT>> serial flowctrl soft 1
Ok!
ROOT>> serial flowctrl hard 2
Ok!
ROOT>> serial flowctrl softhard 3
Ok!
ROOT>> serial flowctrl none 4
Ok!
```

IV.5.5 – Automatic disconnection of a Modem

If there is no data traffic during a determined deplay on port where the modem is connected, MI-ETH can decide to disconnect the modem (DTR signal deasserted).
For example, to disconnect the modem installed on port 3 if this one is not used during 2 mn, use the following command :

```
ROOT>> serial timeout 120 3
```

To disable this option, please set a delay of 0 second.
WARNING !! This automatic disconnection is sometimes managed directly by the modem.

IV.5.6 - Association of a modem form to a port

To use a modem on a port, you have to associate a modem form to this part. A modem form contains configuration parameters for the modem (see IV-8 Management of a modem connection).
To associate a modem form named *"dialing*" to port 1, enter :

```
ROOT>> serial modem dialin 1
```

IV.5.7 – AutoUser mode

AutoUser mode enables MI-ETH to launch automatically a session on the specified port for a certain user. To configure MI-ETH's port 1 in AutoUser mode for a user named *"Paul*", enter:

```
ROOT>> serial autouser paul 1
```

CAUTION !! You must definitely not define any password for the session lauching to be automatical. (see IV.7 - Creation of a user.).

To come back to usual mode on port 1, enter :

```
ROOT>> serial autouser none 1
```

Consult reference manual for more details about commands that are dealt with in this chapter.

IV.6 – Saving and restoring configuration

IV.6.1 – Saving in FLASH memory

All parameters are conserved by MI-ETH in RAM, up to the next MI-ETH boot. If you want to conserve all your parameters, you must write them in FLASH memory. Make this operation with the following command :

```
ROOT>> save
Status: done.

ROOT>>
```

IV.6.2 – Saving a configuration copy

You can save a MI-ETH configuration copy on a remote machine ; thanks to the TFTP (trivial file transfer protocol) protocol. MI-ETH parameters will be sent via TFTP to a file previously created on the remote machine.
Proceed as follows :
Create an empty file on your TFTP server, example with the following UNIX command :

```
echo -n > /tmp/mieth.conf
chmod 666 /tmp/mieth.conf
```

Indicate to MI-ETH the IP address of the server where the file will be staved, as well as the name of this file and start the transfer :

```
ROOT>> system config server 192.168.1.1
ROOT>> system config path "/tmp/mieth.conf"
ROOT>> system config store
```

To restore configuration file, execute the next command :

```
ROOT>> system config load
```

CAUTION !! The loaded configuration is stocked in RAM. Save it in FLASH memory with the **save** command :

```
ROOT>> save
Status: Done.

ROOT>>
```

IV.7 – Opening of a user account

The opening of a user account can be made through the user commands.
Thanks to these commands, you can create or modify a user form.
To create a new form, use the **user add** command. You have to specify fields values according to your needs :

IV.7.1 – Name and password

When you create a user, use the **user add** command, and precise the name and password of this user. Example :

```
ROOT>> user add paul pws25d
```

If you do not need a password for this user (to use Autouser mode for example), replace it by quotations marks.
The **user delete** command delete a user form. Example :

```
ROOT>> user delete paul
```

The **user show** command displays the users list or one particular user's characteristics.

```
ROOT>> user show
User                    Comment
-------------------------------------------------------------------------------------------------------------
paul
User show paul + detail
```

IV.7.2 - Comments

The **user comment** command enables you to insert a comment about the user (his whole name for instance). This comment appears on the right of the screen when you display the users list.

```
ROOT>> user comment paul "Paul Duchemin"
```

IV.7.3 - In and Out Communication

The **user in** and **user out** commands respectively allows the incoming and outcoming connections in asynchronous PPP mode only. Example :

```
ROOT>> user in paul yes
Ok!
ROOT>> user out paul no
Ok!
```

IV.7.4 - Callback

In case of a modem connected on a port in terminal mode (or in PPP mode), MI-ETH can be configurated so that it calls dial-in user back. Thus, phone number dialed by MI-ETH can be predefined, *(static callbac*k), or provided by the user, *(dynamic callbac*k).
In case of a static callback, MI-ETH calls the first number that is specified in the callback list.

```
ROOT>> user callback paul static
```

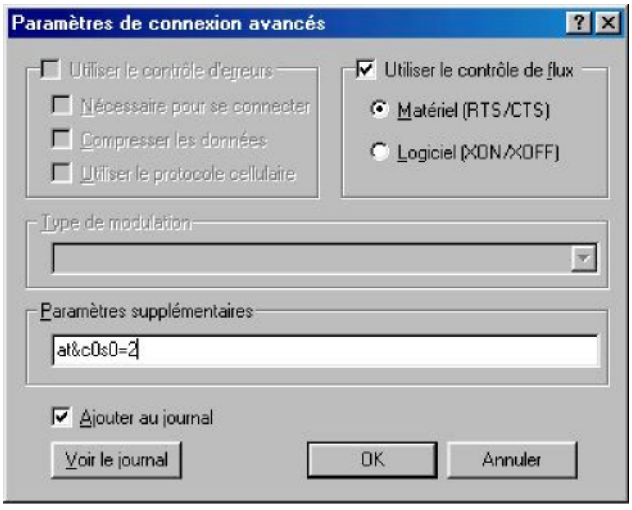
In case of a dynamic callback, MI-ETH asks its number to the dial-in user. For safety reasons, number provided by the client must belong to a determinate list containing not more than three numbers. Neverless, if the "*" character is specified as the first number, every number provided by the user will be accepted.

```
ROOT>> user callback paul dynamic
```

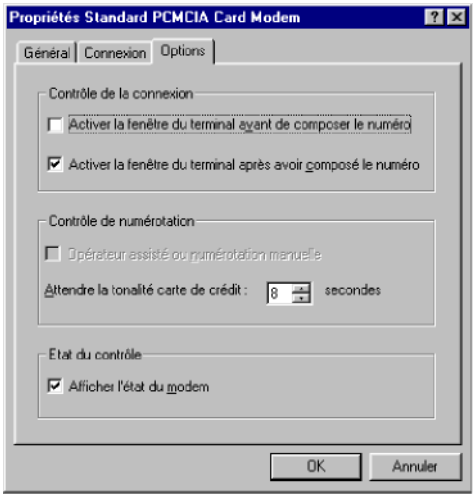
To enter numbers in the callback list, use the following command :

```
ROOT>> user callbacknb paul 1 0240252326
Ok!
ROOT>> user callbacknb paul 2 0251235689
Ok!
```

Call-back between WINDOWS 95/98 and MI-ETH :
In case of a PPP connection from Windows 95/98, at the moment when user is identifying himself to MI-ETH login, MI-ETH will cut current connection in order to call the user back. Consequently, Windows 95/98 PPP connection phase will be interrupted. Indeed, Windows 95 looks after modem DCD state and interrupts the connection if it falls.
One simple solution to this problem is to make the modem DCD always active:
- Configuration panel
- Modems
- Your modem's properties
- Connection tab
- Advanced...
- Supplementary parameters field
- Add : "at&c0s0=2"



You also have to activate terminal window after dialing :
- Your modem's properties
- Options tab



IV.7.5 - Audit

The user audit command enables to record connections achieved by ther. They can be consulted using **audit show login** command. Example :

```
ROOT>> user audit paul yes
```

IV.7.6 - Optional remote IP Address

The Optionnal remote IP address can be configured thanks to **user netaddr** command. In PPP or ISDN mode, (integrated ISDN), MI-ETH assigns this IP address to the user when he connects from a remote machine. If IP address is already attributed, connection is refused.

```
ROOT>> user netaddr paul 192.168.1.50
```

Address' mask can be modified thanks to the **user netmask** command. Example :

```
ROOT>> user netmask paul 255.255.255.0
```

IV.7.7 - Telnet Address

In terminal mode (telnet), MI-ETH will automatically launch a telnet session on a remote machine corresponding to the specified IP address.

```
ROOT>> user telnetaddr paul 1 192.168.1.20
```

If you want to launch telnet session on another port (other than port 23), you just have to enter the following command :

```
ROOT>> user telnetport paul 1 25
```

IV.8 – Management of a modem connection

A Terminal can be connected directly on a MI-ETH port (serial local control) or through modem pair. In this last case, the prompt will be displaged only if the DCD entry signal of the port becomes active **(serial linectrl mode**m). If this signal becomes inactive, all open sessions on this port will be closed.
MI-ETH uses Modem form in which the administrator can specify a list of AT commands to setup a modem.
To create a new modem form, use the **modem add** command. Example:

```
ROOT>> modem add mod33600
```

The modem name is used when you have to associate the modem form to a MI-ETH port.
To display modem forms list or the characteristics of a particular form, use the following commands :

```
ROOT>> modem show
Modem                    Comment
-------------------------------------------------------------------------------------------------------------
dialin
dialout
```

```
ROOT>> modem show mod33600
```

The **modem comment** command enables you to add comments about the modem (its whole name for example) :

```
ROOT>> modem comment mod33600 "modem ACS 33600 bauds"
```

The **init modem** command enables you to change modem initialization sequence. For more informations about the AT commands, consult your modem's documentation. Example :

```
ROOT>> modem init mod33600 at&k3%c3
```

The **modem dial** command enables to define connection sequence of the modem.
Example :

```
ROOT>> modem dial mod33600 atd
```

This form must be associated to the port where the modem is located :

```
ROOT>> serial modem mod33600 1
```

IV.9 - Audit

Audits generate some usefull information to control MI-ETH's activity or to analyse configuration errors. These information are displayed in real time or redirected a file. **Audit** commands enable you to set type and level of captured information, as well as the displaying mode of information.

IV.9.1 - Audit on a remote machine

To create an audit on a remote machine, use the **audit add** command with following parameters :

AUDIT ADD <syslog | trap> <remote IP> <level> <type>

- Syslog / Trap : if you use syslog option , information will be sent to syslogd daemon of the specified remote machine. If your machine does not run syslogd daemon, you must install one.
If you use an smtp administration soft, you can display MI-ETH's audit thanks to the trap option.
- Remote IP : It is the IP address of the remote machine on which you will display MI-ETH's audit.
- Level : It is preciseness level of audit messages. These differents levels are

warning	Minimum (displaying of errors messages).
notice	Like **warning** level but there is messages about MI-ETH's activity.
info	Like **notice** level with more detailled messages about running process.
debug	All existing messages with maximum details.

- Type : It defines type of messages that you want to see in audit. These differents types are

auth	Messages about authentication
printer	Messages about lpd daemon.
Idsn	Messages about isdn connections.
system	Messages about system parameters (DHCP, RADIUS, save, ...).
ppp	Messages about PPP negociation.
async	Messages about asynchronous ports.
net	Messages about network
firewall	Messages about Firewall
all	All preceeding types.

IV.9.2 - Audit on your console

To create audit on your console, use the **audit add console** command in this way :

AUDIT ADD console <level> <type>

Audit will display in your current session's window. You will still be able to enter commands with this shell.
Level and type parameters do not change.

IV.9.3 – Audit in a file

To send auditin a file, use the **audit add buffer** command in this way :

AUDIT ADD buffer <level> <type>

Level and type parameters do not change.
You will be able to show this file thanks to **audit view** and **audit last** commands. The **audit view** command enables you to show the entire audit file.

```
ROOT>> audit view
```

With the **audit last** command, you can display the last lines of the audit file :

```
ROOT>> audit last 10
```

In this example, you will show the ten last lines of the audit file.

IV.9.4 – List of created audits

The **audit show** command enables you to display the list of declared audits. Example :

```
ROOT>>  audit show
buffer        0.0.0.0
                        debug    : async
                        warning : auth ppp system
console       0.0.0.0
                        warning : all
```

IV.9.5 - Delete audit

To delete an audit, use one of the following commands :

AUDIT DELETE <syslog | trap> <remote IP> <level> <type>
AUDIT DELETE <buffer | console> <level> <type>

IV.9.6 – Start / stop audit

When you create an audit with the **audit add** command, it is immediatly actived. But you must restart a console audit after any reboot of your MI-ETH. This can be made thanks to the **audit start** command :

```
ROOT>> audit start
Ok!
```

To stop a console audit, use the audit stop command :

```
ROOT>> audit stop
Ok!
```

V.Terminal Mode (Telnet)

V.1 - Introduction

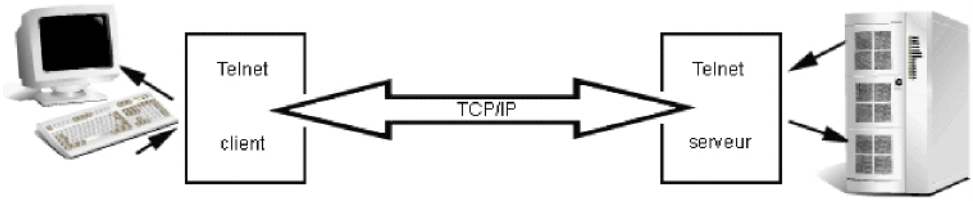
In this mode, MI-ETH affords a login connection as a UNIX system would. This implies a user account and terminal definition notion. It is necessary, for each port defined in the terminal mode :
- to specify the type of terminal used (ANSI, VT100, WYSE, etc...).
- to redefine, if necessary, keys sequences dealing with the management of terminals (EOF, ERASE, KILL).
- to create users accounts.

It is possible to allocate a port to a user (AutoUser mode). In this case, MI-ETH will open a session without asking for a user name.
User will be in a MI-ETH shell like the administrator one, but with a limited choice of commands.
Moreover, up to 3 different sessions can be open on any server. Use the command :

```
ROOT> telnet monserveur
```

Connection login can be immediately displayed on the terminal screen **(serial quick ye**s) or after pressing the touch of the Enter key **(serial quick n**o).
Moreover, MI-ETH has specific functionalities making a remote terminal connection through modems possible.

V.2 – Telnet



Telnet protocol has a "client" part linked to the terminal and a "server" part linked to the network server. These 2 parts communicate through TCP/IP whose physical support can be, for example, an Ethernet LAN.
MI-ETH manages Telnet client part, while server part is managed by the system server.
For example, if the user wants to connect to the server *"myserve*r", he will execute the following command :

```
ROOT> telnet monserveur
```

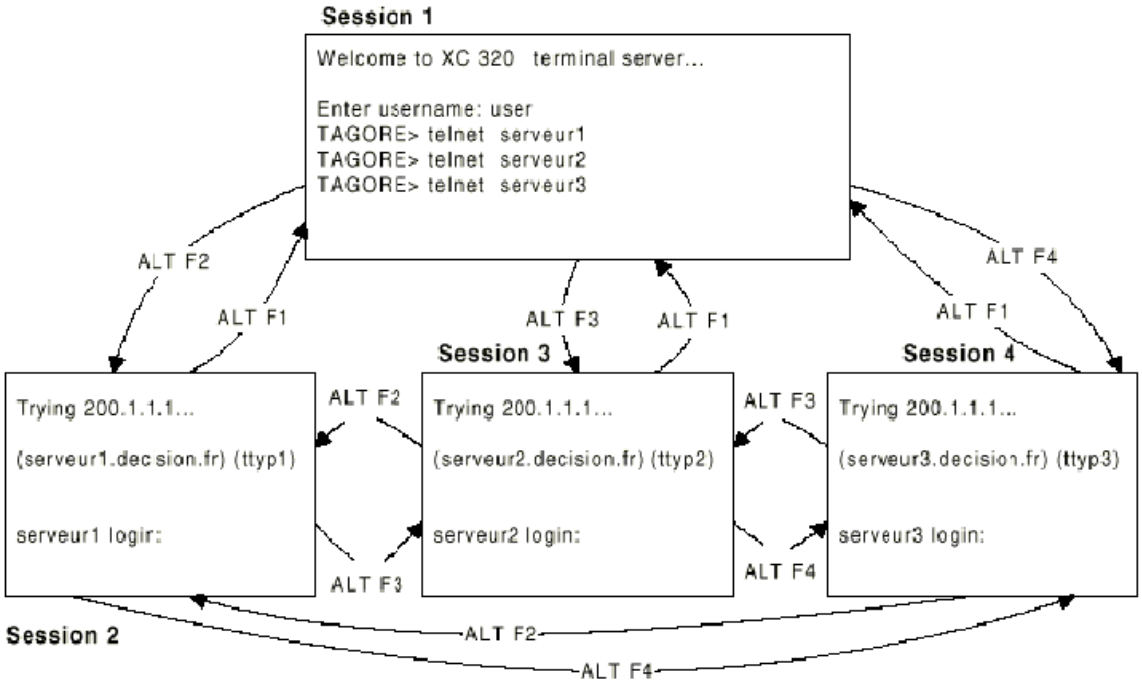
From now on,the Telnet client will ask Telnet server to open a session on the system server.
Then, Telnet client and server are going to converse together, so that all characters entered with the keyboard will be transmitted to the open session on the system server.
As a consequence, the server will return characters to display on the terminal screen.

A Telnet session is open on TCP port 23. If you want to open a session on another port (1301 for example), type the following command :

```
ROOT> telnet monserveur 1301
```

V.3 – Multi-sessions

Up to three Telnet sessions may be issued on one same terminal :



A terminal with an emulation that can bear a change of screen switching (ANSI, WYSE60, etc...) makes a multi-screens between different Telnet sessions possible. It is necessary to setup asynchronous port with :

```
ROOT>> serial termtype "ansi"
```

Note that the specified terminal type (ANSI in this example) is the remote application terminal type negociated during session establishment.
With 'ALT F2', 'ALT F3' and 'ALT F4' control keys you can respectively commute terminal on sessions 2, 3 and 4.
With 'ALT F1' combination you can go back to session 1 in order to execute a new session or to suppress an active session.

It is possible to display the open sessions list :

```
ROOT> serial session 5
Port      Session1        Host1       Session2        Host2       Session3        Host3
-------------------------------------------------------------------------------------------------------
5         closed          0.0.0.0     closed          0.0.0.0     closed          0.0.0.0
```

To close one or several sessions on a port, use the **close session** command. For example, if user connected on port 7 wants to close his first session, he will have to enter :

```
ROOT> close session 1
```

If the same user wants to close all his sessions, he will enter :

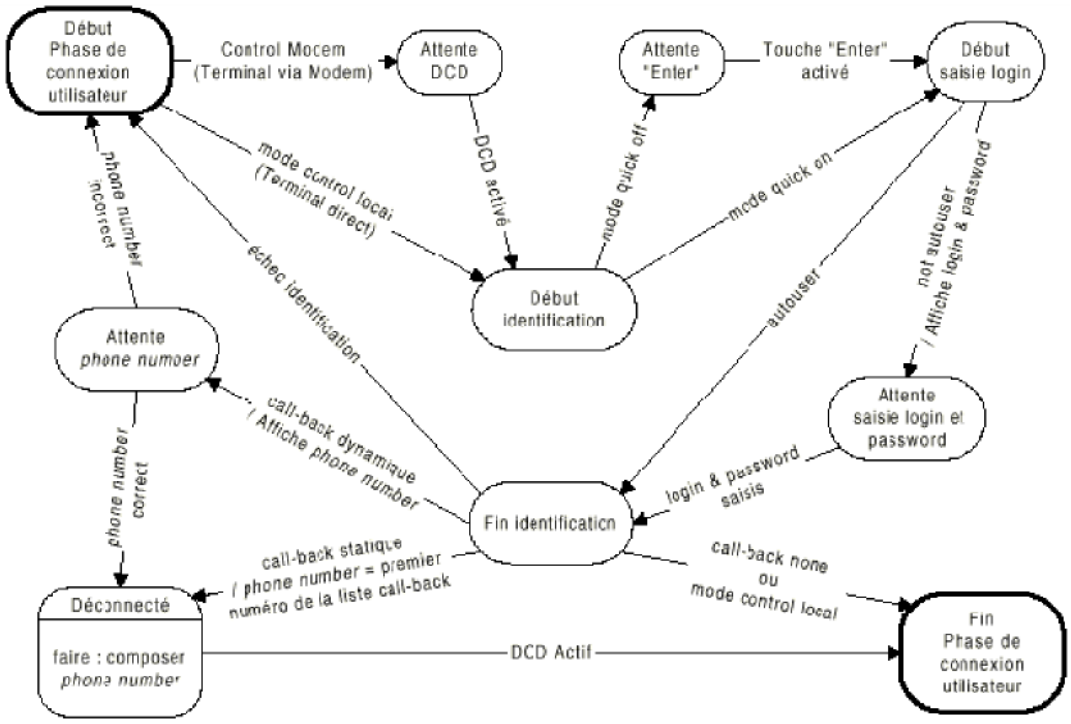
```
ROOT> close session all
```

V.4 – Automatic Telnet

It is possible to automatically launch a Telnet to a certain server system as soon as the user is connected. To achieve this, you just have to add the system IP address to the user form, thanks to the **user telnetaddr** command (see IV.7.7 - Telnet Address).

V.5 – User connection diagram

The user connection scheme can be summarize by the following diagram :



VI. Rawtty and RemoteCOM

VI.1 – Introduction

Rawtty and RemoteCOM enable a server to use an MI-ETH asynchronous port as one of its own ports. To perform this operation, you just have to install a driver on server and to setup a few parameters on your MI-ETH.

VI.2 – RemoteCOM under Windows 95/98 & NT

RemoteCOM is a driver for Windows systems which enables you to redirect COM ports on MI-ETH. RemoteCOM is made to manage flow control signals (DTR, DSR and DCD).

VI.2.1 – Setting up the MI-ETH

Only flow control parameter must be set on the MI-ETH.
Configuration example with Telnet console :

First, display current ports configuration :

```
ROOT>> serial show all
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet Time0
1    raw  9600  soft     local 8  none 1
2001     0
2    raw  9600  soft     local 8  none 1
2002     0
```

If you want to set hard flow control on ports 1 and 2, enter :

```
ROOT>> serial flowctrl hard 1 2
```

Then, reset ports with new parameters :

```
ROOT>> reset port 1 2
```

Display ports configuration again to see if new parameters are well set :

```
ROOT>> serial show all
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet Time0
1    raw  9600  hard     local 8  none 1
2001     0
2    raw  9600  hard     local 8  none 1
2002     0
```

Speed, parity, csize and stop bit are send by RemoteCOM to the MI-ETH.
RemoteCOM transmits DTR signal and recepts states of DCD and DSR signals.

VI.2.2 - Installation of RemoteCOM driver

RemoteCom for Windows is integrated into the intallation of Advanced Adminitrator.

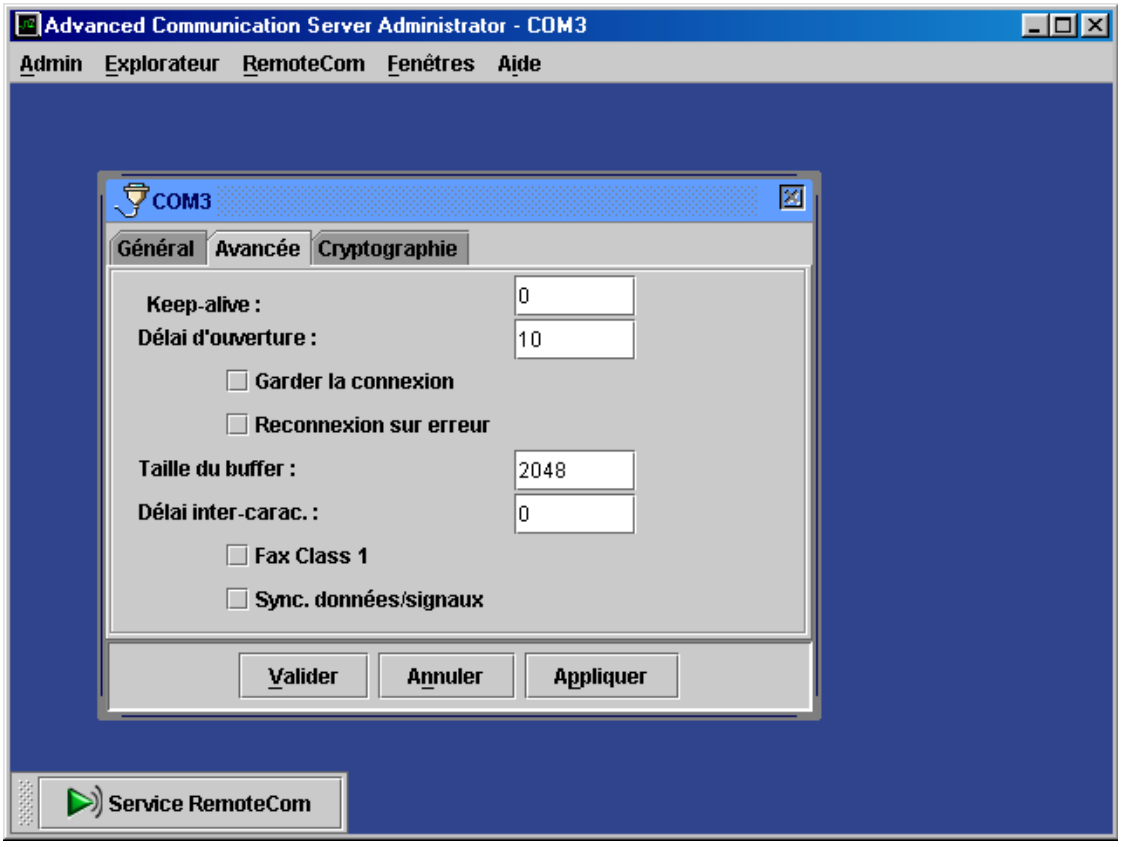
RemoteCOM for WINDOWS is composed of :
- A service module: RemoteCOM(service).

- A driver : RemoteCOM(driver).
- A program for setup, audit and control of RemoteCOM service.

WARNING : Do not mistake RemoteCOM service and RemoteCOM driver (that is also a service). This one is automatically started and stopped by RemoteCOM service.

To create a new port COM on your PC.

- Restart your system.
- In the *"Start*" menu, click on :
- *Programs ->*; *ACS ->*; *Advanced Administrator*.
- Click on *"Edit*" in the main menu.
- Create : select *"Create*".
- Enter MI-ETH's IP address (ex : *192.168.4.23*) , MI-ETH's port number (ex : 7), WINDOWS device (ex : *COM3*) and connection's parameters.
- Change : select *"View&Modify*".
- Delete : select *"Delete*".



Opening timeout : Default parameter is 10 secondes (rarely modified).
WARNING : You must add some timeouts created by the network. Results of the connection trie can come after the open timeout delay.

Keepalive : Enables you to detect any error on the network link.
0 means that KEEPALIVE is stopped. A too small value uselessly overloads the network (15 seconds is a "good" value).

Keep the connection : The network connection will be established at start of RemoteCOM service. The port will not be used by another station.

Restart connection if error : Automatically restart connection (with a delay of 10 seconds) after a disconnection performed by remote MI-ETH. RemoteCOM will try to restart connection every 10 seconds.

Packet mode : Synchronise application that is use RemoteCOM with data flow. The "data transmitted"(EV_TXEMPTY in WaitCommEvent()) label is right only when transmitted data have really been send by the remote port.
The "COMSTAT.cbOutQue" field in ClearCommError() is updated in the same way.
Note : With FAX CLASS 1 modem, this mode (used with a buffer size of 128 bytes) make possible reception and transmission of FAX.

Signals/data synchronised mode : Activate this mode to transmit all signals (TX, RX,RTS,CTS,DTR,DSR,DCD) synchronised with data. You must not activate flow control to use this mode.

Buffers size : 1 to 4096 octets
Default size is 2048 (value 0 or 2048).
In some cases, application seems to have send all data but these data are in buffers. If the application stops the connection before all data have been transmitted, you can reduce buffers size to fix this problem.

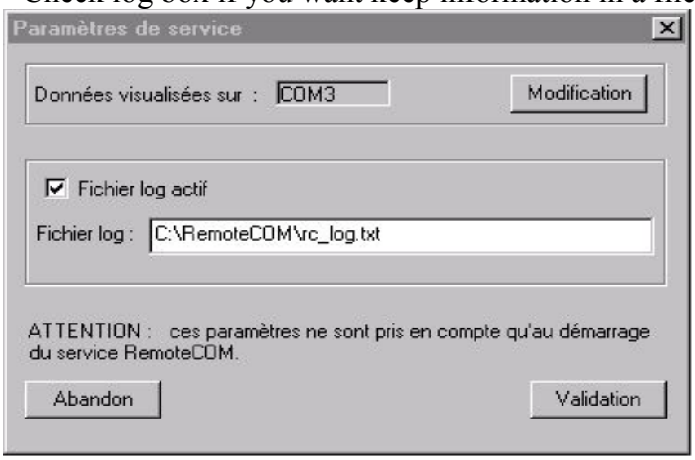
Overhelming of inter-character timeout : 0 to 500 milliseconds.
The Network cut data into several packets. Then, characters of a same "logical" frame (ex: reply message of a modem) can be divide and received with an higher delay than delay of a standard COM port. If you use inter characters timeout option of windows's COM ports, you may have to use this option too.

WARNING : Any modification will be effective after a restart of RemoteCOM service. If RemoteCOM service is already running, you must stop and restart it.
In the main menu, click on "Service", then click on "Start" (or "Stop" and "Start").
Note : When you have just finished RemoteCOM installation, RemoteCOM service is disable.
The first time you will restart your system, RemoteCOM will warn you and then you will be able to choose the activation mode of RemoteCOM service.
You can change service activation mode service at any time with WINNT services control panel.

Displaying RemoteCom service status

Click on Service -> *parameters*

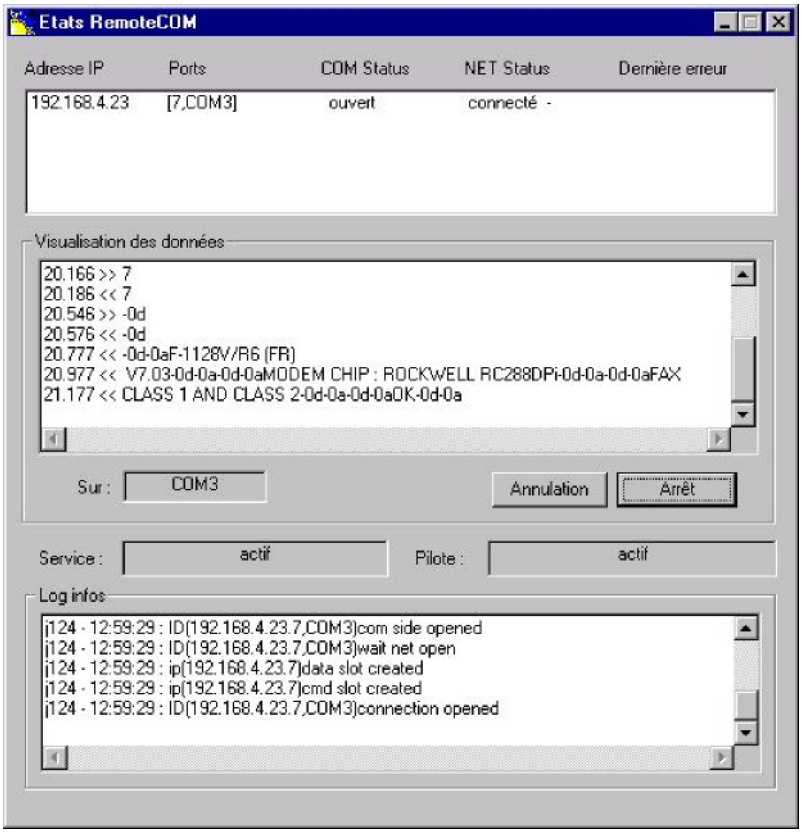
- Select COM port you want display.
- Check log box if you want keep information in a file.



- Be sure that RemoteCOM is not used by any application.
- Click on Service ->*Stop*
- Click on Service ->Start to reset RemoteCOM with new parameters.

Click on Status -> *View in the main menu*

CAUTION: the visualized connections are the last takings into account by the RemoteOM service and not those which could be published previously.



VI.3 - Rawtty

VI.3.1 – Server configuration

Installation example for a rawTTY device driver on UNIX SCO Open Server :

```
#./install.sh

MI-ETH rawTTY installation
adresse IP du MI-ETH ? : 192.168.1.42
>>192.168.1.42 Ok [RET,ÆOÆ] Non [æNÆ] DEL-sortie ? O

PORTS en mode RAW (ex: 1 2 7 (enter) ) ? : 5 6
>> ( 5 6 ) Ok [RET,ÆOÆ] Non [æNÆ] DEL-sortie ? O

relinkage du noyau en cours (...)

Ok

INFO:
- pour la prise en compte des modifications REBOOTez le
systeme
- par defaut les logins sont off
  les noms des ttys sont : /dev/tag <no du port sur MI-
ETH >
  utiliser la commande SCO : enable pour activer les
logins
     ex: enable tag01 tag02 tag07
```

VI.3.2 – Setting up MI-ETH

On MI-ETH, you must setup the corresponding port(s) in *"ra*w" mode:

```
ROOT>> serial mode raw  5 6
ROOT>> serial speed 9600  5 6
ROOT>> serial parity none  5 6
ROOT>> serial csize 8  5 6
ROOT>> serial stopb 1  5 6
ROOT>> reset port  5 6
```

VI.3.3 - rawtty, rawclose

On a UNIX server, with **rawtty** and **rawclose** commands , you can adjust some parameters of the "Rawtty" function.
WARNING !! These two commands must not be used when the Driver has been installed !
The syntax of the **rawtty** command is :

```
rawtty [-keepalive <delay>] [-v1] [-d] [-u username] [-c]
device
MI-ETH _IP port_num
```

Keepalive helps to regularly check MI-ETH's physical presence on the network.
A request is carried out every <delay>; second(s), to check if the connection is still valid. If there is no answer within this time range, connection is cut down.
- ″-*v*1" option enables to use command with MI-ETH versions previous to the 3.2 version.
- ″-d" option enables to enter in debug mode.
- ″-u" option enables to create a device under another user account.
- ″-c" option enables to cancel created device, (ex: /dev/raw1), at the end of the rawtty process.
Two commands enable to increase or to decrease delay of the connected keepalive :

The first command increase keepalive delay with one second.

```
kill -USR1 pid_rawtty
```

The second decrease keepalive delay with one second.

```
kill -USR2 pid_rawtty
```

Example for a UNIX server:

```
rawtty /dev/raw4  192.168.1.12  5 &
```

La The **rawclose** command enables to force to close port declared in "rawtty". Syntax:

```
rawclose server_IP_address port_number
```

Example :

```
rawclose 192.168.1.12 5
```

VII.Mux mode

VII.1 - Introduction

Mux mode enables you to establish an asynchronous link from end to end through an IP network. To make this possible, data (composed by characters and changes of signal's states) are separated into packets. These packets are then transmitted, thanks to TCP or UDP transfert's protocols, from one serial port to the other.

VII.2 – TCP Mux

The TCP protocol warrants retransmission of lost or mistaken data and manages packets. This mode must be used in most of the cases.
To configure this connection, you must set the *'client*' port in mux mode and the *'serve*r' port in raw mode.

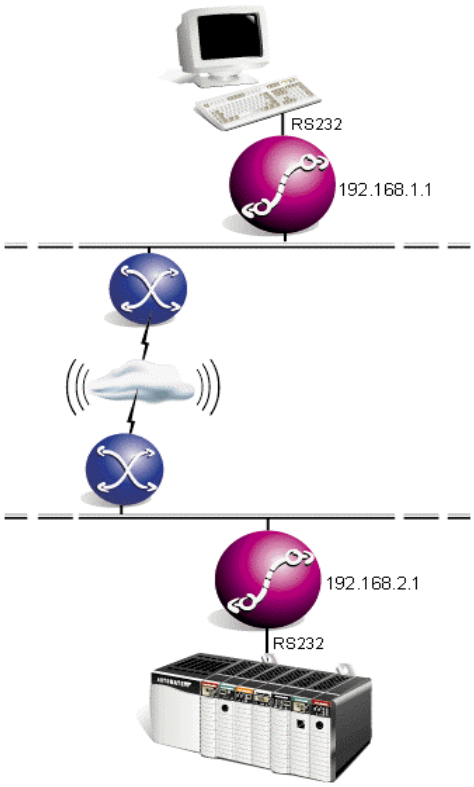
VII.3 – UDP Mux

To transmit the data flow that forwards the asynchronous link from one end of the network to the other, we must change this data flow in a characters flow mixed with codes which match with signal's states. Packaging of data and forwarding of an IP network can generate delays between characters and changes of signal states if they are in different packets. It means that out data flow and in data flow can have mismatching chronogrammes. It can disrupt the good running of some serial transmission's protocols like ones used in industrial environment. Thanks to a measured delay between in data flow and out data flow, UDP mux mode enables most of transmission's protocols to use a connection through an IP network. To configure this connection, you must set both ports in *mux_dg* mode.

VII.4 - Example n°1: Remote management of a PLC by a technician

VII.4.1 - Description

A technician wants to manage a PLC located on a remote TCP/IP network thanks to a terminal (or a terminal emulator).



VII.4.2 - Configuration of client's side MI-ETH

```
ROOT>> serial mode mux 1
Ok!
ROOT>> mux ip 192.168.2.1 1
Ok!
ROOT>> mux port 1 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
------------------------------------------------------------
------------
1    mux  No   Yes   0      192.168.2.1 1    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
------------------------------------------------------------
----------
1    mux  9600  soft     local 8  none 1
2001    0
```

VII.4.3 - Configuration of server's side MI-ETH

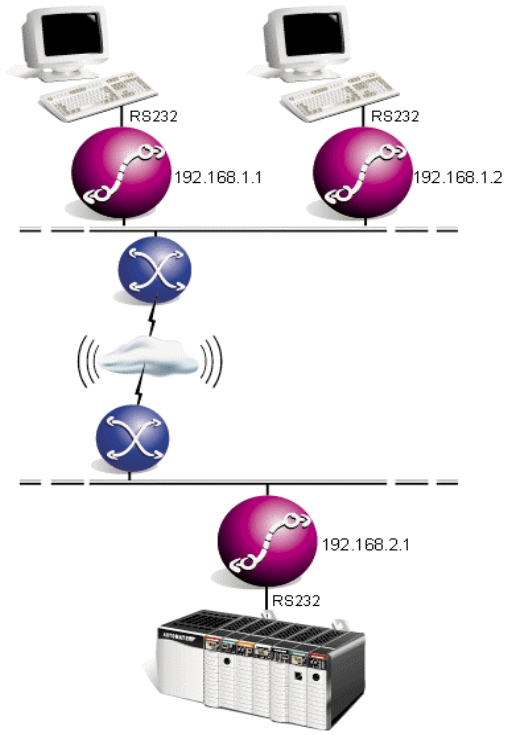
```
ROOT>> serial mode raw 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
------------------------------------------------------------
------------
1    raw  No   Yes   0      0.0.0.0     0    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
------------------------------------------------------------
----------
1    raw  9600  soft     local 8  none 1
2001    0
```

VII.5 - Example n°2: Remote management of a PLC by more than one technician (solution 1)

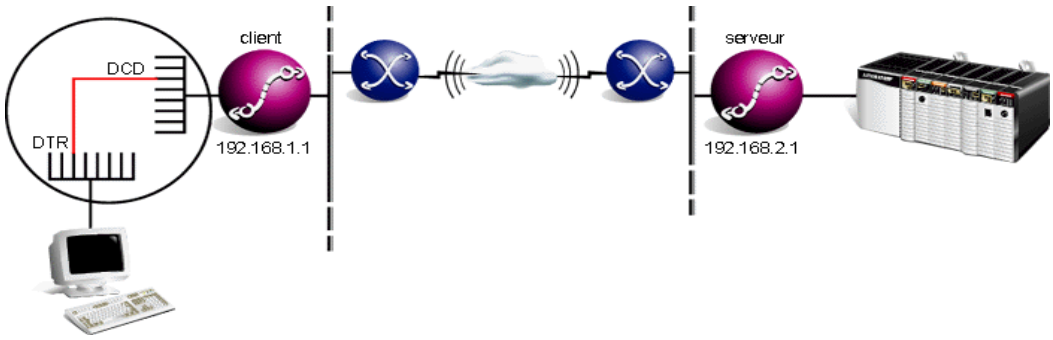
VII.5.1 - Description

Two technicians each have a terminal and they want to manage, one after the other, a PLC located on a remote TCP/IP network.



VII.5.2- Solution

You cannot establish two mux TCP connections to the same destination port at the same time. To solve this problem, we will establish each mux TCP connection only when technician's terminal will be powered.
The technician must power off his terminal to permit the other technician to manage the PLC. When a terminal is powered, there is at least one signal activated (RTS for a terminal or DTR for a terminal emulator). You just have to link this signal to serial port's DCD entry of MI-ETH and to configure management of DCD signal by MI-ETH (serial linectrl modem). TCP connection will be established only if terminal is powered. This connection will be stopped if DCD signal is not on (terminal is off).



VII.5.3 - Configuration of client's side MI-ETH

Make a cable with terminal's RTS (or DTR) signal linked on serial port's DCD signal.

```
ROOT>> serial mode mux 1
Ok!
ROOT>> serial linectrl modem 1
Ok!
ROOT>> mux ip 192.168.2.1 1
Ok!
ROOT>> mux port 1 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
-------------------------------------------------------
1    mux  No   Yes   0      192.168.2.1 1    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
-------------------------------------------------------
1    mux  9600  soft     modem 8  none 1
2001    0
```

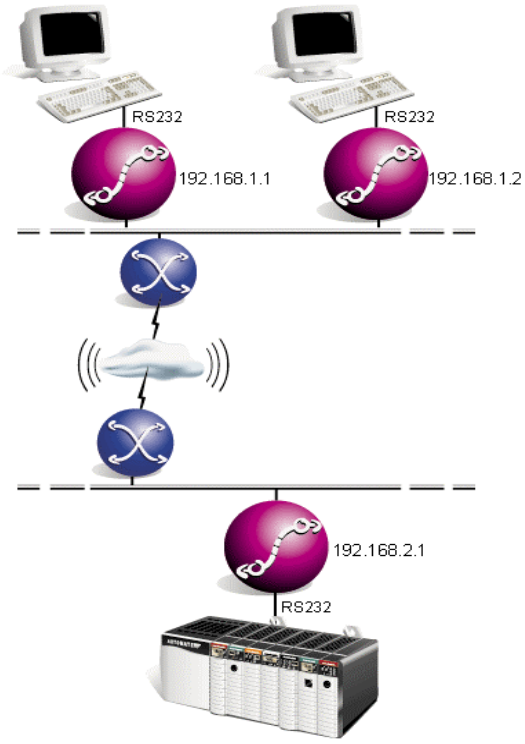
VII.5.4 - Configuration of server's side MI-ETH

```
ROOT>> serial mode raw 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
-------------------------------------------------------
1    raw  No   Yes   0      0.0.0.0     0    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
-------------------------------------------------------
1    raw  9600  soft     local 8  none 1
2001    0
```

VII.6 - Example n°3: Remote management of a PLC by more than one technician (solution 2)

VII.6.1 - Description



VII.6.2 – Solution

The solution above needs that the technician powers off his terminal to allow another technician to manage the PLC. If the first technician forgets to power off his terminal, the PLC is not reachable for the second technician. Another solution is to automatically break the unused connection after a specified timeout.

Connection will be automatically relaunched when the technician will type any character on his keyboard.

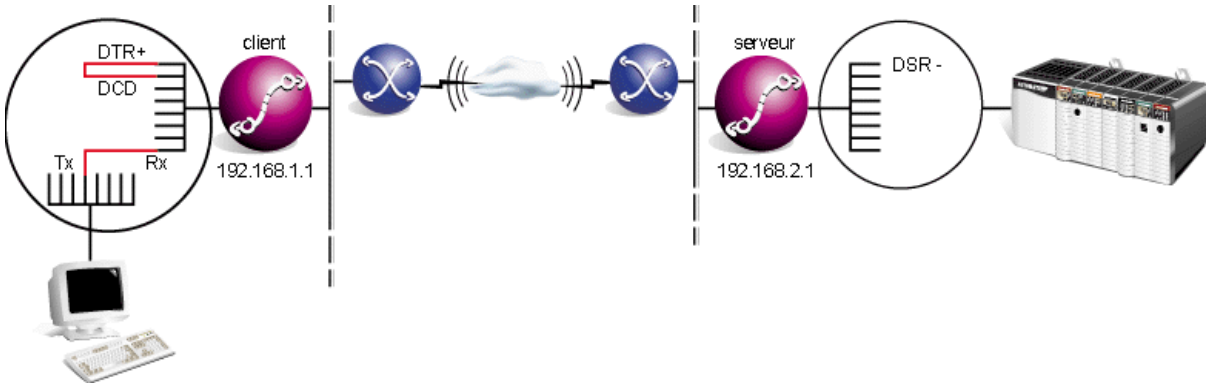
On client's side MI-ETH :

- Link terminal's DTR signal and DCD signal of serial port n°1(make a cable).
- Activate DTR signal on port n°1 **(mux defaultdtr yes** 1).
- Configure management of DCD signal by MI-ETH **(serial linectrl modem** 1).
- Configure MI-ETH to wait a character before launching the connection **(serial quick no**1).
- Specify the timeout delay of the unused connection **(serial timeout** 180 1).

On server's side MI-ETH :

- Be sure that DTR signal state on client MI-ETH will not be changed by DSR signal stateof server MI-ETH **(mux dsrredirect none** 1).

<



Two conditions must be satisfied to establish the connection :

- **DCD signal on** : DTR is on and is linked to DCD of the serial port, the condition will ever be satisfied. But we must be sure that no change of client MI-ETH's DTR signal's state occurs. It is the reason why we stop redirection of DSR signal (mux dsrredirect none 1).
- **Receive a character** : thanks to the serial quick no command, MI-ETH will wait for a character on its serial port before establishing the connection

VII.6.3 - Configuration of client's side MI-ETH

```
ROOT>> serial mode mux 1
Ok!
ROOT>> serial linectrl modem 1
Ok!
ROOT>> serial timeout 180 1
Ok!
ROOT>> serial quick no 1
Ok!
ROOT>> mux defaultdtr yes 1
Ok!
ROOT>> mux ip 192.168.2.1 1
Ok!
ROOT>> mux port 1 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
```

```
------------------------------------------------------------
------------
1    mux  No   Yes   0      192.168.2.1 1    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
------------------------------------------------------------
----------
1    mux  9600  soft     modem 8  none 1
2001    180
```

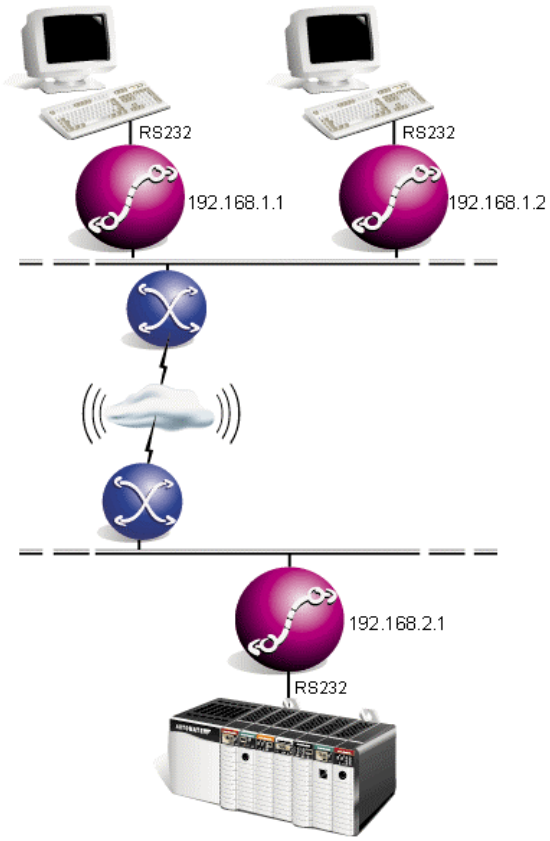
VII.6.4 - Configuration of server's side MI-ETH

```
ROOT>> serial mode raw 1
Ok!
ROOT>> mux dsrredirect none 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
------------------------------------------------------------
------------
1    raw  No   Yes   0      0.0.0.0     0    none   none
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
------------------------------------------------------------
----------
1    raw  9600  soft     local 8  none 1
2001    0
```

VII.7 - Example n°4: Remote management of a PLC by more than one technician (solution 3)

VII.7.1 – Description



VII.7.2 – Solution

The solution above assign the PLC to a technician for a minimum of 3 minutes, even if his terminal is off.
This solution have all advantages of the two preceeding solutions. Access to the PLC will be assigned to the first technician who will try to establish the connection. If the technician power off his terminal, the PLC will be immediatly accessible to the second technician.
One the other hand, if the technician forgets to power off his terminal, the PLC will be accessible to the second technician after 3 minutes.

On client's side MI-ETH :

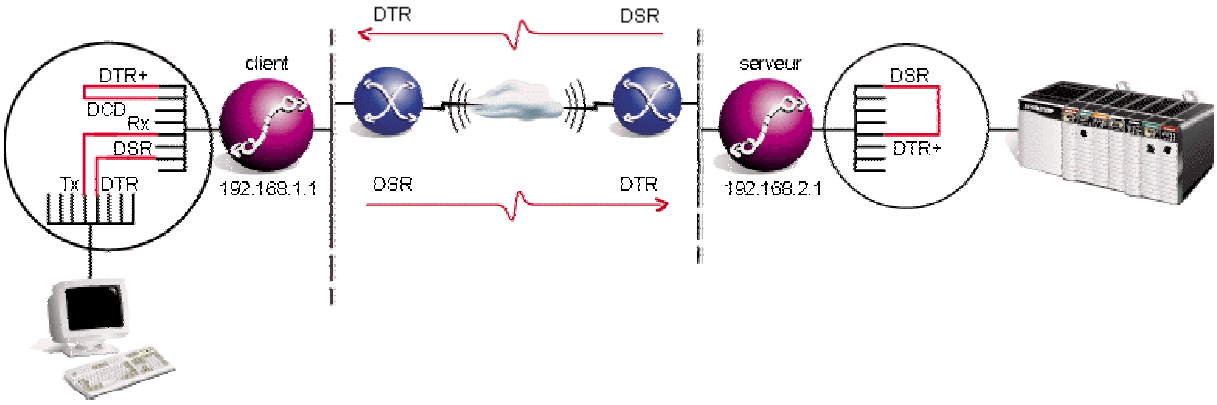
- Link DTR (or RTS) signal of the terminal to DSR signal of serial port n°1 (make a cable).
- Enable redirection of DSR signal to DTR signal of server MI-ETH **(mux dsrredirect dtr** 1).
- Link DTR signal and DCD signal of serial port n°1 (make a cable).
- Activate DTR signal of serial port n°1 **(mux defaultdtr yes** 1).
- Configure management of DCD signal by MI-ETH **(serial linectrl modem** 1).
- Configure MI-ETH to wait a character before launching the connection **(serial quick no** 1).
- Specify the timeout delay of the unused connection **(serial timeout** 180 1).

On server's side MI-ETH :

- Link DTR signal and DSR signal of serial port n°1 (make a cable).
- Activate DTR signal of serial port n°1 (mux defaultdtr yes 1).

- Enable redirection of DSR signal to DTR signal of client MI-ETH (mux dsrredirect dtr 1).

DSR signals of client and server's MI-ETH must absolutly be activated at startup. If DSR signal on client's side MI-ETH is not activated at startup, MI-ETH can't establish the initial connection.
Moreover, an undetermined DSR signal on server's side MI-ETH can force client MI-ETH to break initial connection before this connection can be established.



State of DTR (or RTS) signal on the terminal will be applied to DCD on client MI-ETH. This MI-ETH will be able to establish or break the connection matching with terminal state (on or off).
Others parameters **(serial quick yes** and **serial timeout 180**) allow to start the connection only if MI-ETH is receive a character.

VII.7.3 - Configuration of client's side MI-ETH

```
ROOT>> serial mode mux 1
Ok!
ROOT>> serial linectrl modem 1
Ok!
ROOT>> serial timeout 180 1
Ok!
ROOT>> mux defaultdtr yes 1
Ok!
ROOT>> mux ip 192.168.2.1 1
Ok!
ROOT>> mux port 1 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
-------------------------------------------------------------
```

```
------------
1    mux  No   Yes   0      192.168.2.1 1    none   dtr
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
-------------------------------------------------------------
----------
1    mux  9600  soft     modem 8  none 1
2001    180
```

VII.7.4 - Configuration of server's side MI-ETH

```
ROOT>> serial mode raw 1
Ok!
ROOT>> mux ctsredirect none 1
Ok!
ROOT>> mux defaultdtr yes 1
Ok!
ROOT>> mux show 1
Port Mode Sync Flush KAlive Remote( IP, port) DCD-> DSR->
CTS-> debug
-------------------------------------------------------------
------------
1    raw  No   Yes   0      0.0.0.0     0    none   dtr
none  No

ROOT>> serial show 1
Port Mode Speed FlowCtrl LineC Cs Par. Stop ModemName
Rtelnet TimeO
-------------------------------------------------------------
----------
1    raw  9600  soft     local 8  none 1
2001    0
```

VIII. Printer Mode (LPD)

MI-ETH integrates a LPD spooler to manage several serial printers.
Installing a printer located on MI-ETH, for most UNIX servers, consists simply in adding an entry in the /etc/printcap file.
Example of entry added to the /etc/printcap file :

```
lpACS:lp=:rm=ACS:rp=printer4:
```

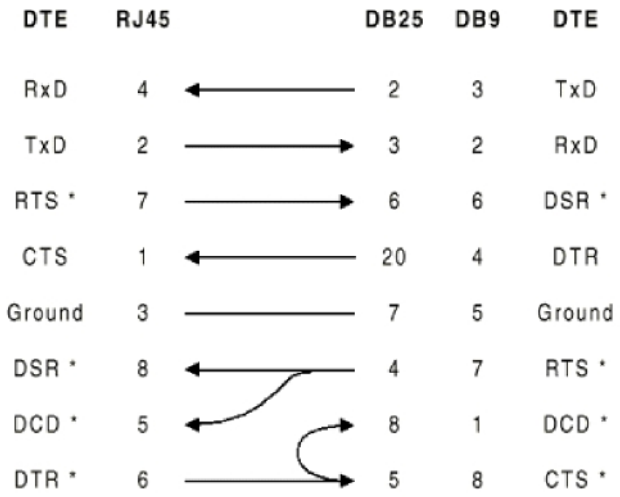
This line means that all prints to the **lpACS** printer, issued by an application running on the server, will be redirected to the printer connected on MI-ETH port 4.
The remote printer name is 'printern' where 'n' is MI-ETH port number on which the printer is connected to.

If a printer is found on the port 4, the configuration will be the following :

```
ROOT>> serial mode printer 4
Ok!
```

In order to control "paperoff" or "offline" signal, you must use Xon/Xoff-RTS/CTS double flow control with the following wiring :

MI-ETH (DTE) toward the asynchronous printer (DTE).
Use the XON/XOFF flow control.
Detection of the printer unavailability (OFF LINE) is possible thanks to the printer DTR.
Signals that are not by a "*" are necessary and sufficient, the ones that are by a "*" are optionnals and correspond to a more standard wiring.



IX. Rtelnet Mode

IX.1 - Definition

With the Rtelnet protocol you can interconnect an equipment without network interface with a TCP/IP network. This equipment can be, for example, an operating system without TCP/IP stack (MOS, PICK, etc...) or a peripheral that can be setup by a serial terminal.
The serial equipment interface is connected to a MI-ETH port in Rtelnet mode. The user connects to this equipment when he executes a Telnet session from a TCP/IP machine

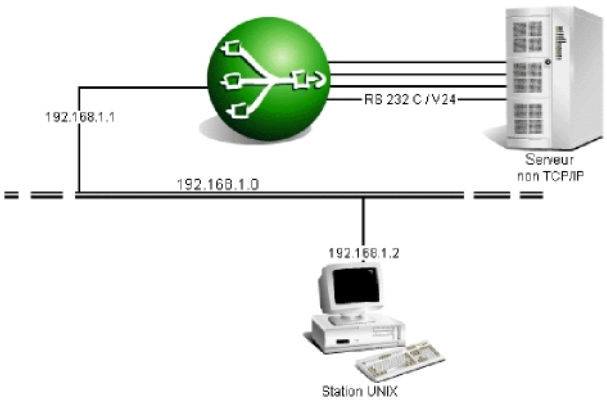


Figure 6 : The TCP/IP workstation connects to the no-TCP/IP server through a Telnet session on MI-ETH

Under UNIX, the command to execute will be :

```
# telnet 192.168.1.1 2001
```

If you have an external modem online on one of MI-ETH port, you could temporarily use the Rtelnet mode to setup it from a remote site.

IX.2 - Configuration

The network port number that is attributed to the Telnet connection is like "20nn"-shaped. "nn" is MI-ETH port number declared in the Rtelnet mode. This declaration is made with the command :

```
ROOT>> serial mode rtelnet 1
ROOT>> reset port 1
```

It is possible to change TCP port number assigned to one MI-ETH port. To achieve this, enter :

```
ROOT>> serial rtelnet 2004 1
```

In our example, port number 2004 is associated to MI-ETH port 1.
To know what TCP port numbers are assigned to MI-ETH ports, enter :

```
ROOT>> serial show all
```

IX.3 - Rtelnet Pool

If you assign the same TCP port number to a group of asynchronous ports, you can create a rtelnet pool. The Telnet client will connect to the first available asynchronous port.
Example :

```
ROOT>> serial rtelnet 2001 1 2 3 4 5 6 7 8
```

X. Asynchrone PPP Mode

X.1 - Introduction

The PPP (Point to Point Protocol) protocol provides a transmission method for IP datagram on serial connections. This enables an interconnection of two remote networks or more simply connection of a remote host to a network.
The serial connection can commute through public network systems (PSTN, ISDN, ...), that is why this protocol provides a double level of security with PAP (Password Authentication Protocol) and CHAP (Challenge Handshake Authentication Protocol).
MI-ETH can accept a PPP connection (Dial-IN). This property gives MI-ETH **a remote access server function**.

X.2 - PAP and CHAP authentication

PAP is similar to the Unix password system. The client introduces itself by providing MI-ETH with a user name and a password. These are compared to the ones defined in MI-ETH user forms.
With CHAP, MI-ETH sends a "challenge" to the client for an authentication, this is a generated random chain of characters sent with its host name. The client has to use the name to find the corresponding code, to combine it with the chain, and to encrypt the chain. The result is sent back to MI-ETH with the client machine name. MI-ETH computes these informations, and in matching case, authorizes the access to the client. Moreover, CHAP sends challenges regularly during the communication to insure that the machine has not been replaced by another.

X.3 - PPP connection setup

MI-ETH uses the concept of PPP form which includes the totality of the necessary parameters for the definition of a PPP connection. If the administrator wants to define, for example, PPP output connections (Dial-OUT) and PPP input connections (Dial-IN), he will create a different form for each of these two configurations.
The creation of PPP forms is detailed in paragraph X.6 - PPP form.

X.4 - Dial-IN connection

The Dial-IN connection steps are :

- A PPP connection request is sent by a remote host.
- The first online free modem answers.
- The modem informs MI-ETH that a connection is active (DCD signal asserted).
- The PPP connection is then established using parameters found in the PPP form associated to the port.

X.5 - - PPP form

You can create , edit or delete a PPP form thanks to the **ppp** commands.

X.5.1 – Creation, cancelling and displaying of a form

The **ppp add** command enables you to create a PPP form. Example :

```
ROOT>> ppp add internet
```

To display PPP forms list or all parameters of a form, use the **ppp show** command.
Example :

```
ROOT>> ppp show
ppp Comment
-------------------------------------------------------------------------------------------------------------------
internet
ROOT>> ppp show internet
```

To delete a form, use the following command:

```
ROOT>> ppp delete internet
```

X.5.2 - Comments

Thanks to the **ppp comment** command, you can add a commentary line to the form.
Example :

```
ROOT>> ppp comment internet "Internet connection form"
```

This line appears when the PPP forms list is displayed **(ppp show**).

X.5.3 - Security

The **ppp security** command manages protection of incoming calls.Three different modes exist:
None : no security.

```
ROOT>> ppp security internet none
```

PAP : MI-ETH requests a PAP authentication from the remote host (see X.2 - PAP and CHAP authentication).

```
ROOT>> ppp security internet pap
```

CHAP : MI-ETH requests a CHAP authentication from the remote host (see X.2 - PAP and CHAP authentication).

```
ROOT>> ppp security internet chap
```

X.5.4 - Authname and password

The ppp authname command enables you to define the user account name that will be used by MI-ETH to authenticate itself to the remote host. Example :

```
ROOT>> ppp authname internet paul
```

Enter the user account password thanks to the **ppp password** command. Example :

```
ROOT>> ppp password internet pws25d
```

X.5.5 - Local IP

Local address of the PPP interface.

If you enter the address *255.255.255.25*5, MI-ETH will accept the address provided by the remote machine.

```
ROOT>> ppp local internet 255.255.255.255
```

If you enter the address *0.0.0.*0, MI-ETH will negotiate its own address, (network interface IP address), with the remote machine.

```
ROOT>> ppp local internet 0.0.0.0
```

X.5.6 - Remote IP

Remote IP address of the PPP interface.

If you enter the address *255.255.255.25*5, MI-ETH will accept address provided by the remote machine.

```
ROOT>> ppp remote internet 255.255.255.255
```

If MI-ETH has already attributed the specified address to another PPP interface, it will negotiate the next available address.

X.5.7 - Proxy ARP

This command define if MI-ETH's ARP table has to include the remote machine IP address (see X.10 - Proxy ARP).

```
ROOT>> ppp proxy internet yes
```

X.5.8 - Mask

It is the network mask of the created ppp interface.

```
ROOT>> ppp mask internet 255.255.255.255
```

X.5.9 - Routing

induce modifications MI-ETH has to carry out on its routing table after a PPP link has been established :

Yes : After a PPP link has been established, MI-ETH will add the next entry to its routing table : destination=Remote IP and mask, bridge=local IP, interface= PPPx.

```
ROOT>> ppp route internet yes
```

Default : After a PPP link has been established, MI-ETH will add the next entry to its routing table : destination=Remote IP and mask, bridge=local IP, interface= PPPx.

```
ROOT>> ppp route internet default
```

No : MI-ETH's routing table is not modified.

```
ROOT>> ppp route internet no
```

X.5.10 - Asyncmap

Enables to transcode, if necessary, one control character or more among the 32 of the ASCII table.If one bit among the 32 of the word Asyncmap is fixed at 1, the rank character that corresponds in the ASCII table will be transcoded.By default, the XON (17), XOFF (19) and GS characters are transcoded. Here is the result : in binary 00100000000010100000000000000000. in hexa. 200A0000. Example :

```
ROOT>> ppp asyncmap internet 200A0000
```

X.5.11 - MRU

Maximum size of IP packets in reception.

```
ROOT>> ppp mru internet 1500
```

X.5.12 - MTU

Maximum size of IP packets in transmission.

ROOT>> ppp mtu internet 1500

X.6 – PPP connection establishment

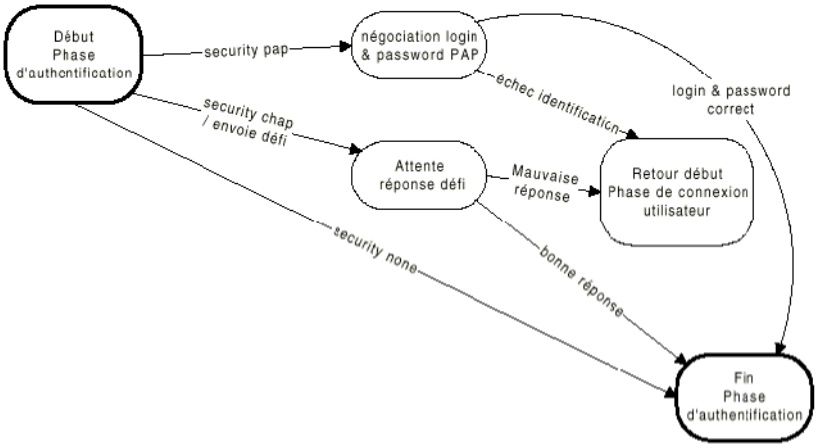
The establishment of an incoming PPP connection proceeds in three steps :

- User connection phase.
- Authentication phase.
- IP address negotiation.

X.6.1 - Dial-IN user connection phase

This phase is the same as the one described in paragraph *V.5 - User connection diagra*m. However, if PAP option is turned on, the user authentication phase (wait login/password) can be interrupted if the calling unit starts directly the PPP negotiation. In this case, the calling unit is identified by the name and the PAP password.

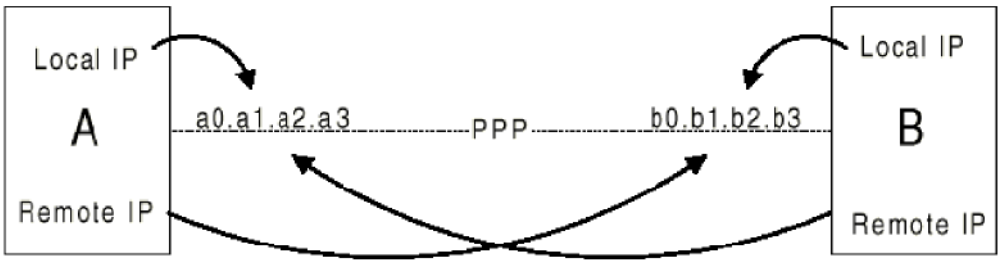
X.6.2 – Authentication phase



Dial-IN : MI-ETH consults its user database to authenticate the calling unit.
Dial-OUT : MI-ETH provides the caller with the name and the password contained in the PPP form.

X.6.3 – IP adress negotiation phase

In this negotiation phase, each machine located at the end of the serial connection sends a request to the other to establish the two IP addresses that will be used during the connection.



To achieve negotiation, the 2 IP address have to be defined by one machine or the other. If the first or the second IP address is not defined, negotiation will fail.
If one of both IP addresses is defined by each of the machines, negotiation result will be undetermined.

Local address :
The local address negotiated by MI-ETH is the one defined in the **Local IP** field of the PPP form.
If Local IP contains *255.255.255.255***,** MI-ETH will accept the address provided by the remote host.
If Local IP contains *0.0.0.*0, MI-ETH will negotiate the IP address of its network interface.

Remote address :
To determine the remote address for negotiation, MI-ETH proceeds in the following order :

- In case of a **Dial-IN** access, it uses in priority IP address specified in the user form (see IV.7.6 - Optionnal remote IP address).
- If the form does not contain an optional remote IP adress, it takes the IP address specified in the Remote IP field from the PPP form.
- If this address is already allocated to another interface, it takes the next free address.

Exemple : Remote IP contient 192.168.5.1	
User A connected	IP Adress 192.168.5.1
User B connected	IP Adress 192.168.5.2
User A disconnected	IP Adress 192.168.5.1 free
User C connected	IP Adress 192.168.5.1

If **Remote IP** contains *255.255.255.255*, MI-ETH will accept the address provided by the remote host.
If **Remote IP** contains *0.0.0.***0,** MI-ETH will negotiate the next free IP address to its network interface.
If **Remote IP** field contains a network address, MI-ETH will accept the address provided by the remote
host (See X.5 - Dial-OUT connection).

WARNING !! The network administrator will have to insure that IP addresses, susceptible to be provided by MI-ETH, are not used by other machines.

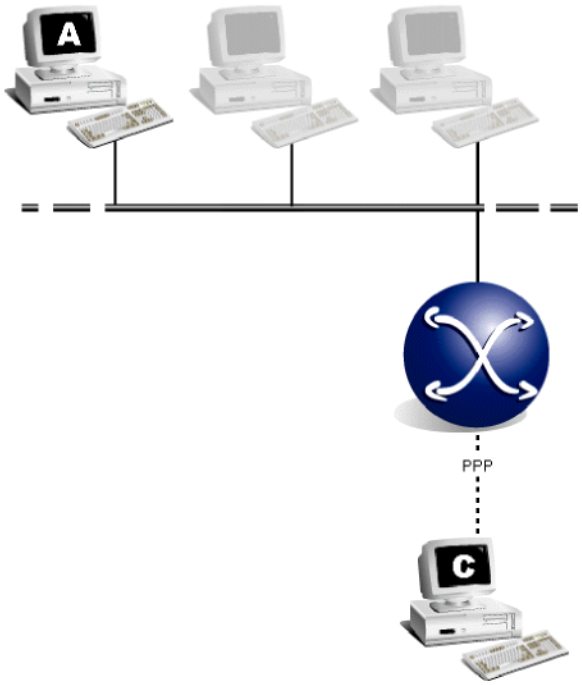
X.7 - Compression

MI-ETH bears BSD compression over a PPP link.
In case of PPP connection through modems between two MI-ETHs, please avoid to active V.42bis compression from modems and the BSD compression for MI-ETH together..

X.8 - Proxy-ARP

In an Ethernet-located TCP/IP network, each machine is both characterized by its IP address and by is MAC address (Medium Access Control). For an IP frame to be lead from one machine to another, you previously have to encapsulate it in an ETHERNET frame. This implies that an A machine dispatching an IP frame to a B machine must necessarily know B' s MAC address. To achieve this, each machine manages an IP and MAC address matching table. This ARP table is dynamically built. Its entries are determined according to the needs and are destroyed after a certain time. If the A machine can' t find in its ARP table an entry giving the B machine' s MAC address, it will make an ARP request to all the network physically present machines. Theoretically, only the machine corresponding to this IP address can react and issue its MAC address to the B machine.

This rule from the ARP protocol, does not apply if for instance, the B machine wants to play the part of a proxy-ARP to the C machine. Examine step by step this mechanism on the following example :

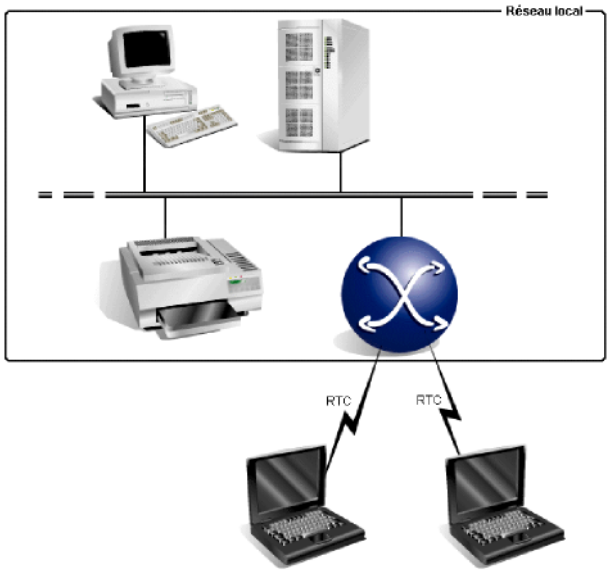


- The A machine wants to emit an IP frame to the C machine.
- A' s ARP table has no entry indicating the C' s MAC address
- A sends a request to all the Ethernet network physically present machines.

- MI-ETH deceives A, by sending its own MAC addresses to it.
- A encapsulates C-assigned IP frame in a MI-ETH-assigned Ethernet frame.
- MI-ETH receives the Ethernet frame and extracts the IP frame.
- As the IP frame is not assigned for it, MI-ETH consults its routing table in order to determine the real addresses.
- MI-ETH observes that address is accessible via its PPP interface.
- MI-ETH encapsulates the C-assigned IP frame in a C-assigned PPP frame.
- C receives the PPP frame and extracts the IP frame first emitted by A.

If one or several C machines are temporarily network connected, proxy-ARP simplifies the IP frames routing. Indeed, all machines of the local network see these remote connection as a local connection.

X.9 - Example n°1 : Remote access server



X.9.1 - Configuration

To setup MI-ETH, please execute the following process :

- Setup the network interface :

```
ROOT>> route add net eth0 192.168.1.0 0.0.0.0 255.255.255.0
Ok!
```

- Specify your DNS address. It will be transmitted to the client when the PPP link will be established :

```
ROOT>> domain server1 < DNS IP address>
```

- Setup asynchronous interfaces :

```
ROOT>> serial speed 115200 1
Ok!
ROOT>> serial mode ppp 1
Ok!
```

```
ROOT>> serial control modem 1
Ok!
```

- Create users account :

```
ROOT>> user add paul xxxx
OK!
ROOT>> user comment paul "Paul Duchemin"
OK!
ROOT>> user in paul yes
OK!
ROOT>> user out paul no
OK!
ROOT>> user audit paul yes
OK!
```

To give a user a particular address, you just have to use the **user netaddr** command :

```
ROOT>> user netaddr paul 192.168.1.50
Ok!
```

To allow MI-ETH to call a particular user back (callback), just choose the *static* option and define the user number in the Callback list field.
Define a modem form if none of those predefined corresponds to your modem :

```
ROOT>> user callback paul static
Ok!
ROOT>> user callbacnb paul 1 0251809000
Ok!
ROOT>> modem add sporster
Ok!
ROOT>> modem comment sporster "Us Robotics Sporster 33600"
Ok!
ROOT>> modem init sporster at&k3%c3
Ok!
ROOT>> modem dial sporster atd
Ok!
```

- Associate the Modem form to the port where the modem is connected :

```
ROOT>> serial modem sporster 1
```

- Define the PPP connection parameters (PPP form) :

```
ROOT>> ppp add ppptest1
Ok!
ROOT>> ppp comment ppptest1 "Test Nb 1"
Ok!
ROOT>> ppp security ppptest1 pap
Ok!
```

```
ROOT>> ppp local ppptest1 0.0.0.0
Ok!
ROOT>> ppp remote ppptest1 192.168.1.46
Ok!
ROOT>> ppp mask ppptest1 255.255.255.255
Ok!
ROOT>> ppp proxy ppptest1 yes
Ok!
ROOT>> ppp route ppptest1 no
Ok!
```

- Save and activate the configuration :

```
ROOT>> save
Status : done
Ok!
ROOT>> reset port 1
Ok!
```

X.9.2 – Configuration of Windows 98/95 Dial-Up PPP Client.

In the *Dial-Up* Networking window, right-click on a connection icon, and then click on Properties. Create a "remote access network" new connection. The "Type of Dial-Up Server" menu in the property window of this connection, should look like this :

Click on *"TCP/IP Setting*s". The DNS IP address will be provided by MI-ETH :
Start remote connection :

X.9.3 – Resources sharing on WINDOWS 95/98/NT

If your LAN is composed by WINDOWS machines, the resources sharing uses the NETBIOS protocol. MI-ETH does not route NETBIOS frames, they must be encapsulated in IP frames. To do this, you must use a **WINS** server or edit *"lmhost*s" files on each LAN machines.

X.10 – Access Server function without Proxy-ARP

The situation is almost the same as previous. However, modify the routing table of the server. A simple solution is to add the next route on the server :

route add -host 192.168.1.46 gw 192.168.1.45 (UNIX system command)

If the remote host connection is unpermanent, this solution is not valid. It is indeed not conceivable to adapt the server routing table to each connection and disconnection of a remote host.
A solution to this problem is to assign to remote hosts an IP address belonging to a sub-network different from that used by the server. For example, if you initiated Remote IP with the value *192.168.2.*1, all machines connected on MI-ETH will be respectively assigned numbers *192.168.2.***1,** *192.168.2.***2,** *192.168.2.*3, etc...Thus, you just have to add the next route on the server :

route add -net 192.168.2.0 gw 192.168.1.45 (UNIX system command)

XI.Advanced fonctions

XI.1 – Firewall

XI.1.1 - Definition

The different network interfaces of MI-ETH (eth0, ppp, isdn) are made to receive and transmit IP frames. The firewall embedded in MI-ETH enables the network administrator to precisely define MI-ETH behavior during reception and broadcast of each frame. The firewall also enables you to define diiferent behavior depending on time and day. The choice can be made in the three following situations :

- The IP frame is MI-ETH-assigned (Input)
- The IP frame is sent by MI-ETH (Output).
- The IP frame pass through MI-ETH (Forward).

In each of these situations and according to the frame type, MI-ETH can behave one of the following ways :

- Accepting the operation (Accept).
- Refusing the operation and advising the sender (Reject).
- Refusing the operation without warning the sender (Deny).

A frame type is characterized by :

- The transmitter.
- The receiver.
- The protocol (UDP, TCP, ICMP).
- Eventually, the port number (telnet, ftp, smtp, ...).

The firewall profile can be achieve in defining chains and policy rules. All parameters are specify in the Firewall page, with the HTTP configuration tool.

XI.1.2 - Chains

A chain enables you to define enviromnent in which rules will be applied. To create a chain, define the chain starting day (or hour), the chain ending day (or hour) and the Firewall type concerned. While the chain is activated, the rules it contains are used by MI-ETH.

To select the starting day (or ending day) of a chain, just fill some of the following fields : Year, month, day, day of the week, hour, minute. You are not obliged to fill all fields. For example, if you just define the starting hour and th ending hour, this chain will be activated every day during this time range.

XI.1.2.1 – Examples

Chains and rules [edit chain] [menu]

Name	Type	Start of time range						End of time range						State
		Hr	Mn	D/w	D	M	Y	Hr	Mn	D/w	D	M	Y	
example1	forward	*	*	monday	*	*	*	*	*	friday	*	*	*	Waiting

Edit chain example1 [edit rule] [edit redir.] [view chains] [menu] Add Delete Cancel OK

Enable	Type of IP frames	Comments
☑	forward	

--Time range activity--

	Hour	Minute	Day	Month	Year	Day of week
Start	*	*	*	*	*	monday
End	*	*	*	*	*	friday

Chain activated every week from Monday to Friday. Just select, "**Start Day of the Week** *: Monday*" and "**End Day of the Week** *: Friday*".

Chains and rules [edit chain] [menu]

Name	Type	Start of time range						End of time range						State
		Hr	Mn	D/w	D	M	Y	Hr	Mn	D/w	D	M	Y	
example2	forward	*	*	*	1	*	*	*	*	*	15	*	*	Waiting

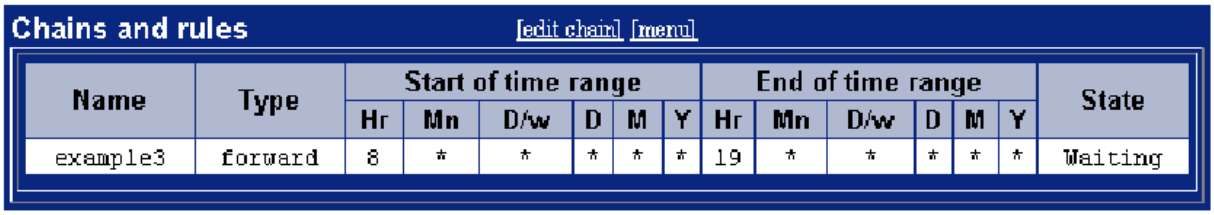
Edit chain example2 [edit rule] [edit redir.] [view chains] [menu] Add Delete Cancel OK

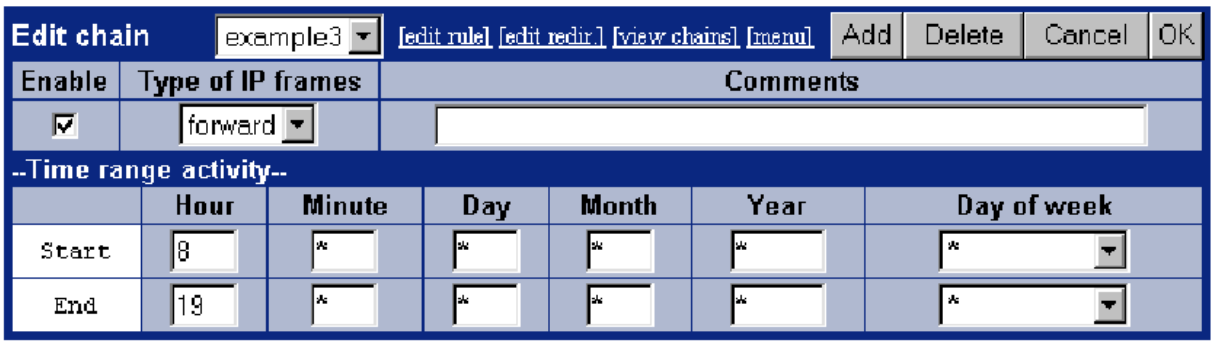
Enable	Type of IP frames	Comments
☑	forward	

--Time range activity--

	Hour	Minute	Day	Month	Year	Day of week
Start	*	*	1	*	*	*
End	*	*	15	*	*	*

Chain activated every month from the 1st to 15th. Just fill these two fields, "**Start Day** *: 1*" and "**End Day** *: 15*".


Chains and rules
[edit chain] [menu]
Name
Type
Start of time range
End of time range
State
Hr Mn D/w D M Y Hr Mn D/w D M Y
example3 forward 8 * * * * * 19 * * * * * Waiting



Chain activated every day from 8AM to 7PM. Just fill these two fields, **"Start Hour** : 8" and **"End Hour** : *1*9".

To create a chain, you must precise the chain's Firewall type (Input, Output or Forward). Select the Firewall type in the menu.

XI.1.3 - Rules

These rules are made of a premise the characterizes a frame type and of a conclusion that specifies the kind behavior to be adopted.

Premise	>	Conclusion
sender AND target AND protocol AND port	>	accept OR reject OR deny

(AND : logical and ; OR : logical or)

MI-ETH looks for the first rule that characterizes the IP frame to be analyzed. If it finds none, a default **policy** is adopted.

WARNING !! : MI-ETH looks for the first rule that characterizes the IP frame. Order of rules is very important. Enter rules in bad order may forbid any access to MI-ETH.

A rule is applied to an IP frame if the following conditions occur :

- IP-sender AND Mask = Source
- IP-target AND Mask = Destination.
- The protocol matches the one indicated in the rule.
- Eventually, port matches the one(s) indicated in the rule.

There is two other parameters that you must define when you create a rule :

XI.1.3.1 – Bidirectionnal option

The Bidirectionnal option create a reciprocal rule. It is equivalent as the creation of a second rule with target value of the first rule as source and source value of the first rule as target.

WARNING !! This option is valide only in a Forward Firewall chain.

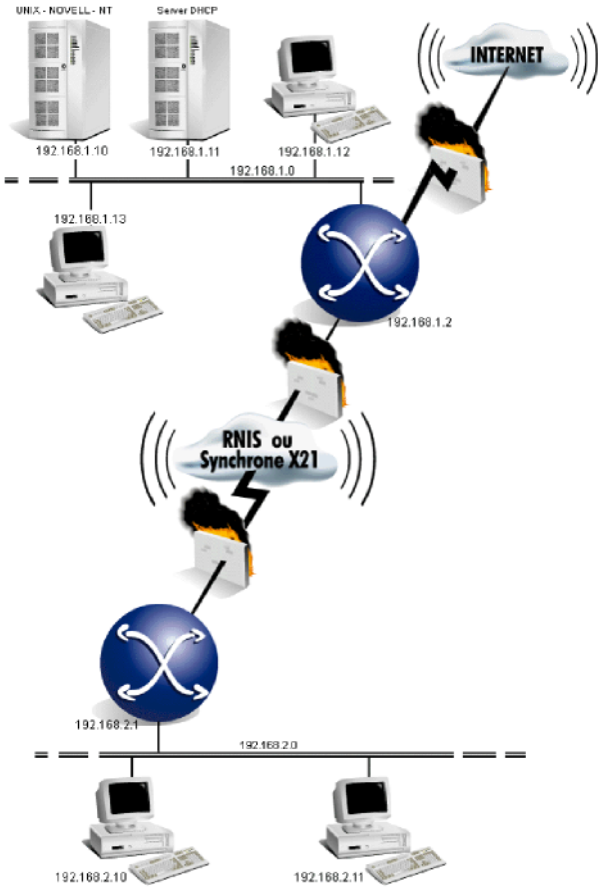
XI.1.3.2 - Type Of Service (TOS)

There are five Type of Service :

- Minimum Delay.
- Maximum Throughput.
- Maximum Reliability.
- Minimum Cost.
- Standard.

These services allows, in few cases, to optimize connections. Most of the time, standard option is good enough.
Mostly used configurations are **Minimun Delay** for Telnet connections and **Maximum throughput** for data transfert with FTP.

XI.1.4 – Examples



This figure will act as a support for examples described below :

XI.1.4.1 - Forward Firewall

For example, you can forbid a particular LAN machine all connections to Internet or forbid a machine to use some services (FTP, mail, web). To forbid these connections, just say to MI-ETH not to route frames coming from the machine to the selected services.
It can be made with a Forward Firewall rule.

1 - Create an active chain in Forward Firewall mode.

Enter the chain name ("chain" for example) in the Name field And select **Forward** in the menu.

Chains and rules [edit chain] [menu]

Name	Type	Start of time range						End of time range						State
		Hr	Mn	D/w	D	M	Y	Hr	Mn	D/w	D	M	Y	
example1	forward	*	*	monday	*	*	*	*	*	friday	*	*	*	Running...

Routing rules

	Address	Netmask	Ports	Bi-dir	Proto.	Policy	T.O.S.
src	192.168.0.0	255.255.0.0			*	accept	Standard
dst	192.168.0.0	255.255.0.0					
src	192.168.2.11	255.255.255.255			*	reject	Standard
dst	0.0.0.0	0.0.0.0					
src	192.168.1.11	255.255.255.255			*	reject	Standard
dst	0.0.0.0	0.0.0.0	www				
src	192.168.1.12	255.255.255.255			*	reject	Standard
dst	0.0.0.0	0.0.0.0	smtp pop-3				
src	192.168.1.13	255.255.255.255			*	reject	Standard
dst	0.0.0.0	0.0.0.0	telnet ftp				
src	192.168.0.0	255.255.0.0			*	masquerade	Standard
dst	0.0.0.0	0.0.0.0					

2 - To create the second rule of this example, please execute the following process :

MI-ETH default behavior is to accept all connections **(default policy** = accept).
In the *Address* field of the **Source** line, enter the IP address of the LAN machine that will not be able to connect to Internet, then enter the mask value in the *Mask* field of the **Source** line.
To join Internet, MI-ETH uses the default route, therefore enter the value *0.0.0.0* in the *Address* field of the **Destination** line and the value *0.0.0.0* in the *Mask* field of the **Destination** line.
In the **Protcol** field, enter the filtred frames type (all in this case).
In the **Policy** field, enter MI-ETH policy for these frames.
And then, keep **Bidirectionnal** box unchecked and **TOS** to standard.
In this example, all frames **(Protocol** = *all*) coming from the *192.168.2.11* machine **(Source** = *192.168.2.1*1) and targeting an Internet host **(Destination** = *0.0.0.*0) will be rejected **(Policy** = *reject*).

To specify a particular port in a rule, just enter its number in the ports field. You can also type the service name in the ports **field** (Ex : *telnet to 23*).
In the *Ports* field of the **Source** line, enter the user plage range that will be forbid (or autorised) by the rule. To do this, you must enter the first and the last number separed by a ":" (Ex : *1024:65535*).
In the *Ports* field of the **Destination** line, enter the port(s) number(s) (or service(s) name(s)) that will be autorised or forbid by the rule. To enter more than one port, just separate them by a space (Ex : *telnet ft*p).
For example, the *192.168.1.11* machine will not be able to connect to a web site **(Ports** = www) but it be able to send e-mails or to make ftp transfert.
As well, the *192.168.1.12* machine will not be able to send *(smt*p) or receive *(po*p) e- mails. Other services can be used. The *192.168.1.13* machine cannot issue a telnet session, nor a FTP transfert.
The last rule allows all *192.168.0.0* subnetwork's machines to connect to Internet.

XI.1.4.2 - Input Firewall

For security reasons, we can expected that an Internet host could not issue a Telnet session or access to a Rawtty port on MI-ETH. Only the LAN machines will be authorized to administrate MI-ETH or to connect to a Rawtty port. To do this, make this **Input Firewall** chain.

Chains and rules [edit chain] [menu]

Name	Type	Start of time range						End of time range						State
		Hr	Mn	D/w	D	M	Y	Hr	Mn	D/w	D	M	Y	
example1	forward	*	*	monday	*	*	*	*	*	friday	*	*	*	Running...

Routing rules

	Address	Netmask	Ports	Bi-dir	Proto.	Policy	T.O.S.
src	192.168.0.0	255.255.0.0			*	accept	Standard
dst	192.168.1.1	255.255.255.255					
src	0.0.0.0	0.0.0.0			*	reject	Standard
dst	192.168.1.1	255.255.255.255	telnet raw				

XI.1.4.3 - Output Firewall

If you want that a terminal connected to MI-ETH cannot issue a Telnet session on any Internet machines, make this **Output Firewall** chain.

Chains and rules [edit chain] [menu]

Name	Type	Start of time range						End of time range						State
		Hr	Mn	D/w	D	M	Y	Hr	Mn	D/w	D	M	Y	
example1	forward	*	*	monday	*	*	*	*	*	friday	*	*	*	Running...

Routing rules

	Address	Netmask	Ports	Bi-dir	Proto.	Policy	T.O.S.
src	192.168.1.1	255.255.255.255			*	accept	Standard
dst	192.168.0.0	255.255.0.0					
src	192.168.1.1	255.255.255.255			*	reject	Standard
dst	0.0.0.0	0.0.0.0					

XI.2 – Radius

XI.2.1 - Introduction

Radius is a client-server protocol aiming at centralizing the users authentication on a dedicated server (see RFC 2138).
To use Radius terminology, we will call MI-ETH NAS (Network Access Server).
The obvious interest of this protocol is the fact that it enables to manage all the accesses to an information site with only one users database, and whatever the type and number of NAS may be.
In addition to its authentication function, a RADIUS server also enables to rule the connection type and parameters, (PPP, telnet, admin, asynchronous or/and RNIS access, connection time...), according to the user who wish to connect.
An extension of the Radius protocol, (RFC 2139), also enables to count the connections.
You can thus keep a track of the connections made by all the users (name, connection length, amount of informations transmitted and received, cause of the end of session).
The working process of MI-ETH integrated Radius client is the following : A user authentication request, (login, PAP, CHAP), is first locally analysed. If MI-ETH contains a suitable user account, connection type and parameters are managed by MI-ETH.
When the session starts, MI-ETH sends a starting signal to the RADIUS server (START).
When the session ends, MI-ETH transmits an ending signal (STOP).
If MI-ETH cannot manage to authenticate the user, the request is transmitted to the RADIUS server. If RADIUS accepts the request, it will transmit the connection type and parameters to MI-ETH. Later, informations about beginning and end of session are transmitted to RADIUS.

XI.2.2 - Attributs Radius supportés par le MI-ETH

Attribute	Definition	Notes
User-Name	Name of the user to identify.	Maximum size : 16 alphanumerical characters
User-Password	User password	Maximum size : 8 alphanumerical characters
CHAP-Password	Challenged user answer emitted by the NAS Size :	128 bits
NAS-IP-Address	NAS IP address	
NAS-Port	Asynchronous port or ISDN channel number	
NAS-Port-Type	Port type	Asynchronous or ISDN
Service-Type	Service type	Login, Framed, Callback Login, Admin, NAS Prompt, Callback NAS prompt
Framed-Protoco	Frame type	PPP only
Framed-IP-Address	IP address attributed to the user	
Login-IP-Host	Telnet server IP address	Callback Login and Login service
Login-TCP-Port	Telnet server port number	Callback Login and Login service
Login-Service	Kind of service used to connect the user with an host machine	Telnet only
Reply-Message	Message to be transmitted to the user	
Callback-Number	Callback number	Implanted only on asynchronous ports
Acct-Status-Type	Session starting or ending informations	START or STOP
Acct-Input-Octets	Number of bytes received during a session	
Acct-Output-Octets	Number of bytes transmitted during a session	
Acct-Session-Id	Session identifier	
Acct-Session-Time	Session duration	
Acct-Terminate-Cause	Session ending cause	User Request, Lost Carrier,...

XI.2.3 – Radius requests

To authenticate a user, the NAS emits a request ("Access Request") toward the Radius server. The Radius server can reject the request ("Access Reject") or accept the request ("Access Accept").
If the request is accepted, the NAS can inform the Radius server about the starting and ending of the session ("Accounting Request").
Radius requests carry several attributes. The following table shows the requests emitted by the NAS.

Attribut	Access Request	Account Request "Start"	Account Request "Stop"	Note
User-Name	1	1	1	
User-Password	0-1	0	0	
CHAP-Password	0-1	0	0	
NAS-Port	1	1	0	
NAS-Port-Type	1	1	0	
Service-Type	0-1	0-1	0	
Framed-Protocol	0-1	0-1	0	=1 if Service-Type = Framed or Callback Framed
Framed-IP-Address	0-1	0-1	0	=1 if Service-Type = Framed or Callback Framed
Login-IP-Host	0-1	0-1	0	=1 if Service-Type = Login or Callback Login
Login-TCP-Port	0-1	0-1	0	=1 if Service-Type = Login or Callback Login
Login-Service				=1 if Service-Type = Login or Callback Login
Acct-Input-Octets	0	0	1	
Acct-Output-Octets	0	0	1	
Acct-Session-Id	0	1	1	
Acct-Session-Time	0	0	1	
Acct-Terminate-Cause	0	0	1	User Request, Lost Carrier,...

XI.2.4 - Configuration of thr Radius client on MI-ETH

To activate the authentication Radius client, you just have to give MI-ETH the RADIUS server IP address. A nil value ("*0.0.0.*0") stop the Radius client.
This setting up can be done, either with a navigator (System page), or in command line :

SYSTEM RADIUS AUTHENTIC <server node name | IP address>

In the same way, you can activate the accounting Radius client like this :

SYSTEM RADIUS ACCOUNTING <server node name | IP address>

TCP ports attributed to the Radius server are by default 1645, (authentication), and 1646, (accounting). These two port numbers have been modified in the last two RFC about RADIUS. These new values are 1812 and 1813. If your Radius server takes these new numbers into account, modify them with the following commands :

SYSTEM RADIUS AUTHPORT <TCP port No. of authentic. server>
SYSTEM RADIUS ACCTPORT <TCP port No. of accounting server>

When MI-ETH emits a request to the Radius server, it waits no more than 3 seconds for the server's answer. You can modify this duration with the commands :

SYSTEM RADIUS AUTHTIMEOUT <authentic. server request timeout>
SYSTEM RADIUS ACCTTIMEOUT <accounting server request timeout>

If MI-ETH has no answer at the end of the Time-Out, it repeats its request three times. You can modify this number with the following command :

SYSTEM RADIUS AUTHRETRIES <authentic. server request retries>
SYSTEM RADIUS ACCTRETRIES <accounting server request retries&gt

The Radius protocol cyphers the passwords so that they do not appear clearly in the requests IP frames. You must therefore necessarily specify the coding key (secret). It must be the same as the one of the Radius server.

SYSTEM RADIUS SECRET "key"

XI.3 - DHCP

XI.3.1 - Definition

DHCP is a client-server protocol that aimed at a dynamical attribution of the IP addresses to the machines of a network.
At the starting point, each network machine sends a request to the DHCP server and is given an IP address. MI-ETH integrates a DHCP client that enables it to get an IP address from a DHCP server.

A problem occurs at the networks interconnection: the DHCP requests are frames that do not go through the routers. Consequently, for two connected networks to use the same DHCP server, the router must play the part of a DHCP relay; that is to say that it must pick the DHCP requests up and transmit them back to the DHCP server.
MI-ETH from the network *192.168.2.0* will transmit the DHCP requests from the two machines back to the DHCP server *(192.168.1.*1). Both interconnected networks can thus use the same DHCP server.

XI.3.2 - DHCP client

MI-ETH holds a DHCP client function. If a DHCP server is active on your network, MI-ETH can receive its IP address and many others parameters from the DHCP server. In factory configuration, MI-ETH boots with the DHCP mode active. The system **dhcp client mode** command enables you to select the running mode of DHCP client.

DHCP CLIENT MODE <dhcp | bootp | none>

The **bootp** mode enables MI-ETH to act as a BOOTP client (Bootstrap Protocol). This protocol, which is the predecessor of DHCP protocol, supports less options than DHCP. To deactivate MI-ETH's DHCP client, use the **none** option.

MI-ETH does not know the DHCP server IP address. It broadcast is request on the LAN to join the DHCP server. If you want to specify a particular DHCP server IP address, just use the **system dhcp server** command.
Example :

```
ROOT>> dhcp client server 192.168.10.1
```

If you want to go back to the broadcast method, enter the following command :

```
ROOT>> dhcp client server 255.255.255.255
```

When MI-ETH is booting (if the DHCP client is active), it sends its DHCP request on the LAN and waits for a reply. If it does not receive a reply after 30 seconds, MI-ETH assumes that the request fails. Then, MI-ETH use IP address and other parameters saved in flash memory. To modify the reply timeout, use the **dhcp clent timeout** command.
Example :

```
ROOT>> dhcp client timeout 60
```

To visualize the current parameters of your customer DHCP, enter:

```
ROOT>> dhcp client show
```

WARNING !! To validate DHCP commands, you must save new parameters in flash memory. The next time MI-ETH will boot, it will use these new parameters.

XI.3.3 – DHCP relay

MI-ETH holds the function of DHCP relay. To activate this function, you just have to use the **system dhcp relay** command. This syntax command is the following :

SYSTEM DHCP RELAY <server 1| server2> <IP address>

If the IP address of your DHCP server is 192.168.1.1, the following command will activate MI-ETH's DHCP relay :

```
ROOT>> dhcp relay server1 192.168.1.1
```

Note that you can enter IP address of a second DHCP server. In this case, MI-ETH etransmit DHCP requests to both servers and will take care of the first reply that it will receive.

To launch the DHCP relay, enter :

```
ROOT>> dhcp relay enable yes
```

To stop the DHCP relay, enter:

```
ROOT>> dhcp relay enable no
```

XII.Technical specifications

MI-ETH 2 ports

MI-ETH 2	
LAN ETHERNET	
10 base T	Yes
ASYNCHRONOUS	
Number of ports	2
RS232C Interface	Yes
RS422/485A Interface	No
Overvoltage	15KV ESD
Signals	XON/XOFF,RTS/CTS,DTR,DSR,DCD
Speed	440 kbd
Cabling	Sub D 9 Male
ROUTING	
Static routing	Yes
SECURITY	
PAP, CHAP	Yes
Radius client	Yes

User database	Yes
Automatic callback	Yes
Embadded Firewall	Yes
Time range	Yes
HTTPS SSLV2-V3	Yes (3)
Secured Remote COM	SSL V3-DES 40 or 56 bits,RC4 40 or 128 bits, RSA 512 ou 1024 bits
ADMINISTRATION	
HTTP	Yes
SNMP MIBII, MIB MI-ETH	Yes (2)
Java	Yes
Relay, DHCP client	Yes (2)
Telnet Console	Yes
Serial ports (I / O)	Yes
MISCELLEANEOUS	
Protocols	IP, TCP, UDP, ICMP, ARP, Finger, TFTP, Telnet, Rtelnet, DHCP, BOOTP, HTTP, SNMP, Syslog
PPP	Yes
LPD	Yes
RemoteCOM	Option
GENERAL	
Memory	8 Mo
Timer WatchDog	Yes
Power Supply	207 VAC to 253 VAC, 50 to 60 Hz, 6.5 VA
Size	102,5 x 175 x 47 mm

(1) : Or
(2) : Basical version available
(3) : Basical version available without HTTPS

MI-ETH 4/8 ports

MI-ETH 4/8	
LAN ETHERNET	
10 base T	Yes
ASYNCHRONOUS	
Number of ports	4 or 8
RS232C Interface	Yes
RS422/485A Interface	Yes (Option)
Overvoltage	15KV ESD
Signals	XON/XOFF,RTS/CTS,DTR,DSR,DCD
Speed	440 kbd
Cabling	RJ 45
ROUTING	
Static routing	Yes
SECURITY	
PAP, CHAP	Yes
Radius client	Yes
User database	Yes
Automatic callback	Yes
Embadded Firewall	Yes
Time range	Yes
HTTPS SSLV2-V3	Yes (3)
Secured Remote COM	SSL V3-DES 40 or 56 bits,RC4 40 or 128 bits, RSA 512 ou 1024 bits
ADMINISTRATION	
HTTP	Yes
SNMP MIBII, MIB MI-ETH	Yes (2)
Java	Yes
Relay, DHCP client	Yes (2)
Telnet Console	Yes
Serial ports (I / O)	Yes
MISCELLEANEOUS	
Protocols	IP, TCP, UDP, ICMP, ARP, Finger, TFTP, Telnet, Rtelnet, DHCP, BOOTP, HTTP, SNMP, Syslog
PPP	Yes
LPD	Yes
RemoteCOM	Option

GENERAL	
Memory	16 Mo
Timer WatchDog	Yes
Power Supply	85 VAC to 264 VAC, 47 to 63 Hz, 7 VA
Size	280 x 170,5 x 40 mm

((1) : Or
(2) : Basical version available
(3) : Basical version available without HTTPS

XIII.Cabling

XIII.1 - SUBD9M connector on MIETH 2 ports

Pin	Circuits
1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	N.C

XIII.2 - RJ45 connector on MIETH 4 & 8 ports

Pin	Circuit
1	CTS
2	TxD
3	Ground
4	RxD
5	DCD
6	DTR
7	RTS
8	DSR

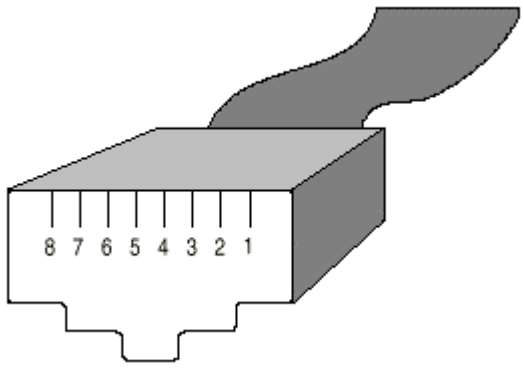
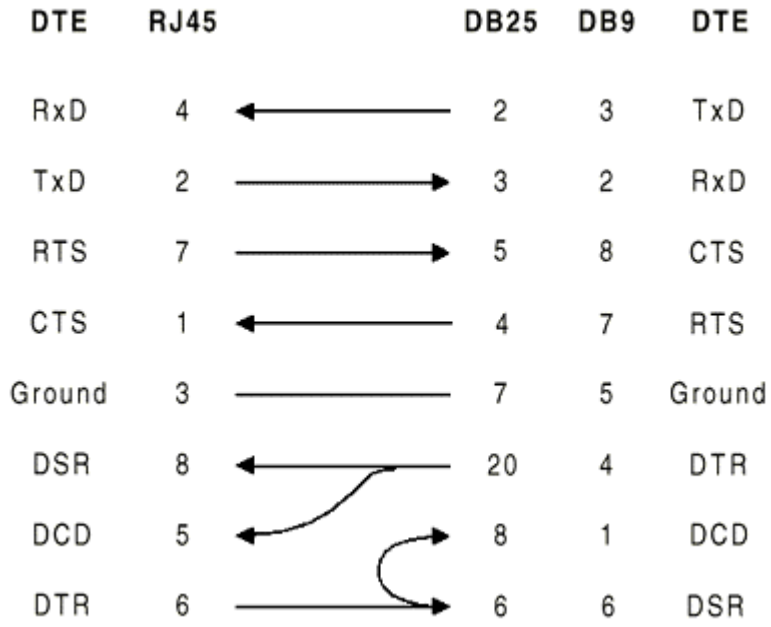


Figure 3 - RJ45 front view

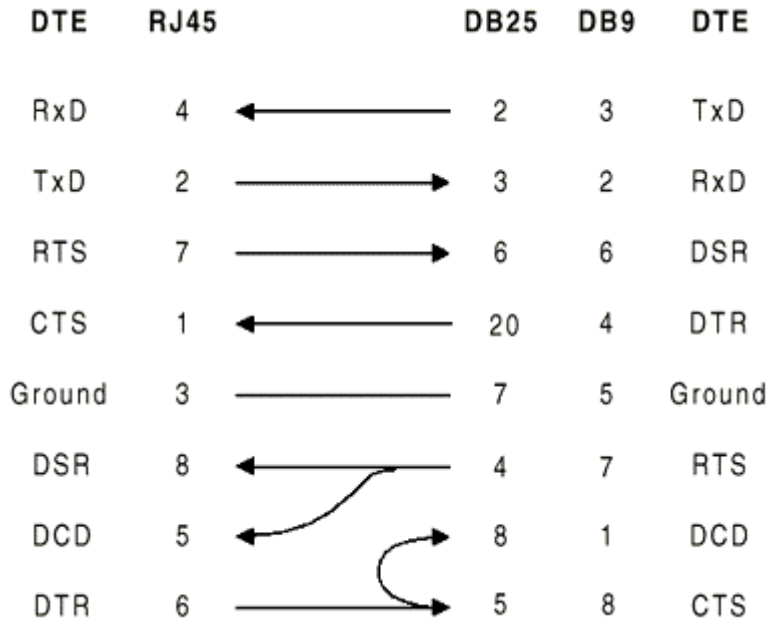
XIII.3 - MI-ETH (DTE) to DTE

With RTS/CTS flow control

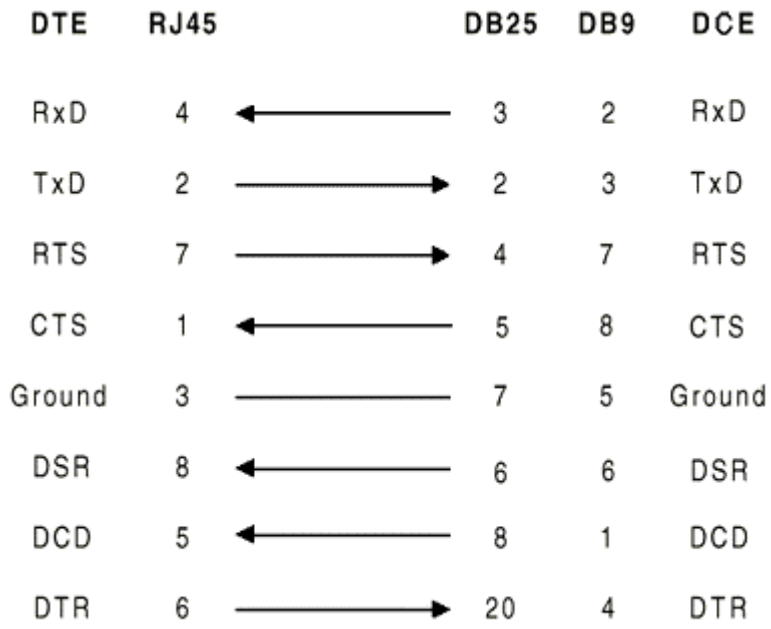


XIII.4 - MI-ETH (DTE) to asynchronous terminal (DTE)

With RTS/CTS flow control

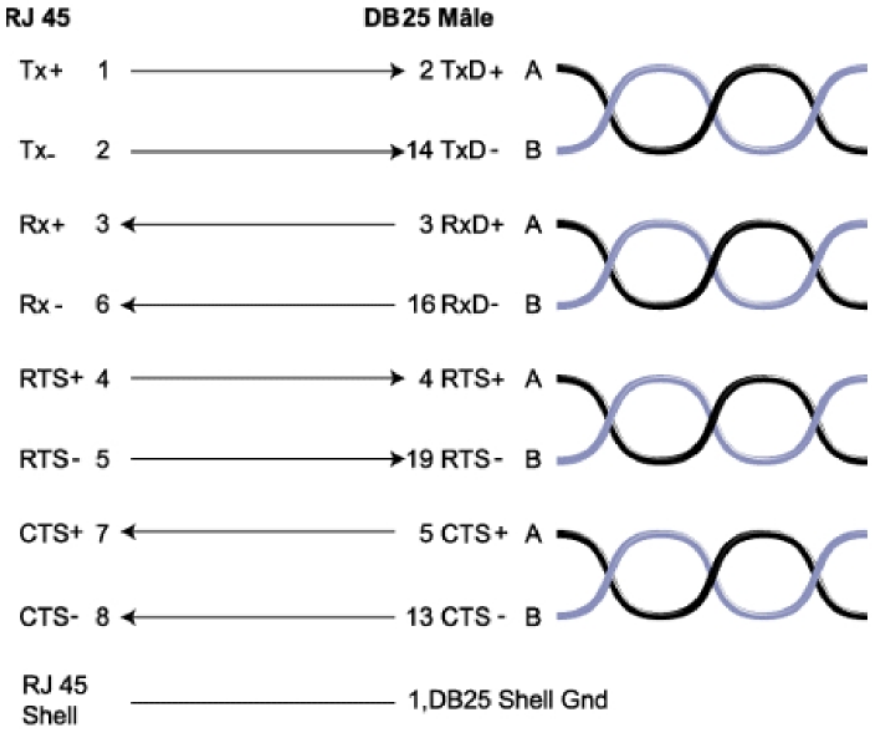


XIII.5 - MI-ETH (DTE) to DCE equipment - RS232C (Modem)

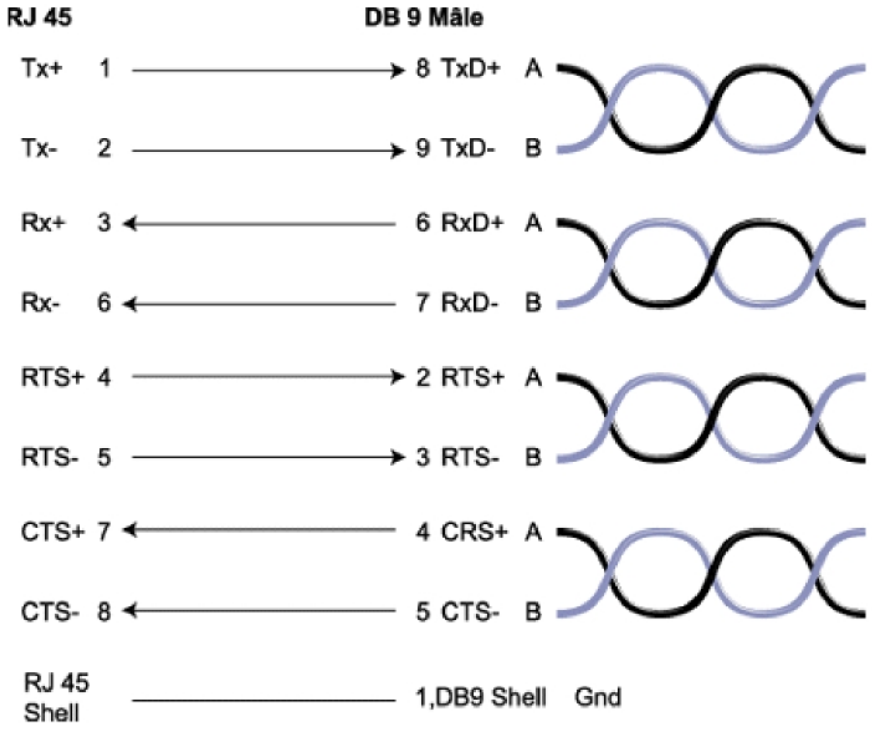


RJ45 - DB25 Male - Comx 232
RJ45 - DB9 Male - Comx 232

XIII.6 - MI-ETH (DTE) to DCE equipment - RS422A

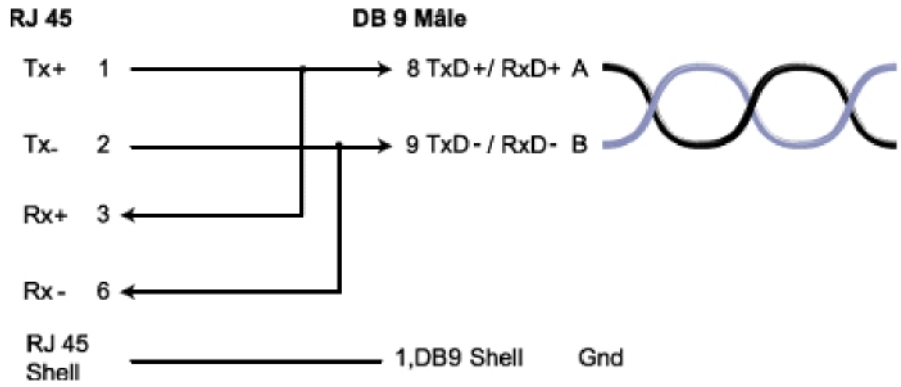


RJ45 - DB25 Male - Comx 422



RJ45 – DB9 Male - Comx 422

XIII.7 - MI-ETH (DTE) to DCE equipment - RS485



RJ45 – DB9 Male - Comx 485